INTELLI 2021

The Tenth International Conference on Intelligent Systems and Applications

ISBN: 978-1-61208-882-2

July 18 – 22, 2021

Nice, France

INTELLI 2021 Editors

Gil Gonçalves, University of Porto, Portugal

Carsten Behn, Schmalkalden University of Applied Sciences, Germany

INTELLI 2021

Forward

The Tenth International Conference on Intelligent Systems and Applications (INTELLI 2021) continued a series of events on advances towards fundamental, as well as practical and experimental aspects of intelligent systems and applications.

The information surrounding us is not only overwhelming, but also subject to limitations of systems and applications, including specialized devices. The diversity of systems and the spectrum of situations make it almost impossible for an end-user to handle the complexity of the challenges. Embedding intelligence in systems and applications seems to be a reasonable way to move some complex tasks form user duty. However, this approach requires fundamental changes in designing the systems and applications, in designing their interfaces and requires using specific cognitive and collaborative mechanisms. Intelligence becomes a key paradigm and its specific use takes various forms according to the technology or the domain a system or an application belongs to.

We take here the opportunity to warmly thank all the members of the INTELLI 2021 technical program committee, as well as all the reviewers. The creation of such a high quality conference program would not have been possible without their involvement. We also kindly thank all the authors who dedicated much of their time and effort to contribute to INTELLI 2021. We truly believe that, thanks to all these efforts, the final conference program consisted of top quality contributions. We also thank the members of the INTELLI 2021 organizing committee for their help in handling the logistics of this event.

INTELLI 2021 Chairs

INTELLI 2021 Steering Committee
Carsten Behn, Schmalkalden University of Applied Sciences, Germany
Stefano Berretti, University of Firenze, Italy
Leo van Moergestel, HU University of Applied Sciences Utrecht, Netherlands
Gil Gonçalves, University of Porto, Portugal

INTELLI 2021 Industry/Research Advisory Committee
David Greenhalgh, University of Strathclyde, Glasgow, UK
Paolo Spagnolo, National Research Council, Italy
Luca Santinelli, ONERA Toulouse, France
Lars Braubach, Hochschule Bremen, Germany
Maiga Chang, Athabasca University, Canada
Chin-Chen Chang, Feng Chia University, Taiwan

INTELLI 2021 Publicity Chair
Mar Parra, Universitat Politecnica de Valencia, Spain
Alvaro Liebana, Universitat Politecnica de Valencia, Spain

INTELLI 2021
Committee

INTELLI 2021 Steering Committee
Carsten Behn, Schmalkalden University of Applied Sciences, Germany
Stefano Berretti, University of Firenze, Italy
Leo van Moergestel, HU University of Applied Sciences Utrecht, Netherlands
Gil Gonçalves, University of Porto, Portugal

INTELLI 2021 Industry/Research Advisory Committee
David Greenhalgh, University of Strathclyde, Glasgow, UK
Paolo Spagnolo, National Research Council, Italy
Luca Santinelli, ONERA Toulouse, France
Lars Braubach, Hochschule Bremen, Germany
Maiga Chang, Athabasca University, Canada
Chin-Chen Chang, Feng Chia University, Taiwan

INTELLI 2021 Publicity Chair
Mar Parra, Universitat Politecnica de Valencia, Spain
Alvaro Liebana, Universitat Politecnica de Valencia, Spain

INTELLI 2021 Technical Program Committee
Azizi Ab Aziz, Universiti Utara Malaysia, Malaysia
Ari Aharari, SOJO University, Japan
Bilal Ahmad, University of Warwick, UK
Zaher Al Aghbari, University of Sharjah, UAE
Sarra Alqahtani, Wake Forest University, USA
Rachid Anane, Coventry University, UK
Zahra Ebadi Ansaroudi, University of Salerno, Italy
Olugbenga Moses Anubi, Florida State University, USA
Benjamin Aziz, University of Portsmouth, UK
Arvind Bansal, Kent State University, USA
Suzanne Barber, The University of Texas at Austin, USA
Carmelo Bastos-Filho, University of Pernambuco, Brazil
Rafael Batres, Tecnologico de Monterrey, Mexico
Carsten Behn, Schmalkalden University of Applied Sciences, Germany
Fayçal Bensaali, Qatar University, Qatar
Lyes Benyoucef, Aix-Marseille University, France
Giuseppe Berio, IRISA | Université de Bretagne Sud, France
Stefano Berretti, University of Firenze, Italy
Jonathan Berrisch, University of Duisburg-Essen, Germany
Mahdis Bisheban, National Research Council Canada (NRC), Canada
Francisco Bonin Font, University of the Balearic Islands, Spain
Frederic Bousefsaf, LCOMS | Université de Lorraine, France
Lars Braubach, Hochschule Bremen, Germany
Simeon C. Calvert, Delft University of Technology, Netherlands

Longzhi Yang, Northumbria University, UK
Leila Zemmouchi-Ghomari, Ecole Nationale Supérieure de Technologie, ENST, Algiers, Algeria

Copyright Information

For your reference, this is the text governing the copyright release for material published by IARIA.

The copyright release is a transfer of publication rights, which allows IARIA and its partners to drive the dissemination of the published material. This allows IARIA to give articles increased visibility via distribution, inclusion in libraries, and arrangements for submission to indexes.

I, the undersigned, declare that the article is original, and that I represent the authors of this article in the copyright release matters. If this work has been done as work-for-hire, I have obtained all necessary clearances to execute a copyright release. I hereby irrevocably transfer exclusive copyright for this material to IARIA. I give IARIA permission or reproduce the work in any media format such as, but not limited to, print, digital, or electronic. I give IARIA permission to distribute the materials without restriction to any institutions or individuals. I give IARIA permission to submit the work for inclusion in article repositories as IARIA sees fit.

I, the undersigned, declare that to the best of my knowledge, the article is does not contain libelous or otherwise unlawful contents or invading the right of privacy or infringing on a proprietary right.

Following the copyright release, any circulated version of the article must bear the copyright notice and any header and footer information that IARIA applies to the published article.

IARIA grants royalty-free permission to the authors to disseminate the work, under the above provisions, for any academic, commercial, or industrial use. IARIA grants royalty-free permission to any individuals or institutions to make the article available electronically, online, or in print.

IARIA acknowledges that rights to any algorithm, process, procedure, apparatus, or articles of manufacture remain with the authors and their employers.

I, the undersigned, understand that IARIA will not be liable, in contract, tort (including, without limitation, negligence), pre-contract or other representations (other than fraudulent misrepresentations) or otherwise in connection with the publication of my work.

Exception to the above is made for work-for-hire performed while employed by the government. In that case, copyright to the material remains with the said government. The rightful owners (authors and government entity) grant unlimited and unrestricted permission to IARIA, IARIA's contractors, and IARIA's partners to further distribute the work.

Table of Contents

Intelligent Task Assignment in Industry 4.0 Production Processes Utilizing Fuzzy Sets

Gregor Grambow, Daniel Hieber and Roy Oberhauser

Dept. of Computer Science
Aalen University
Aalen, Germany
e-mail: {gregor.grambow, daniel.hieber, roy.oberhauser}@hs-aalen.de

***Abstract*—Industry 4.0 production comprises complicated highly automated processes. However, human activities are also a crucial component of these processes, e.g., for machine maintenance. Task assignment of human resources in this domain is challenging, as many factors have to be taken into account to ensure effective and efficient activity execution and satisfy special conditions (like worker safety). To overcome the limitations of current Business Process Management (BPM) Systems regarding activity resource assignment, this contribution provides a BPM-integrated approach that applies fuzzy sets for activity assignment. Our findings suggest that this approach can be easily applied to complex production scenarios, while providing efficient performance even with a large number of concurrent activity assignment requests. Additionally, our evaluation shows its potential for improved work distribution which can lead to cost savings in Industry 4.0 production processes.**

Keywords*—*Business Process Management Systems; Business Process Modelling; Staff Assignment Algorithms; Assignment Automation; Fuzzy Logic.

I. INTRODUCTION

"Industry 4.0" stands for the fourth industrial revolution driven by digitalization [1]. Highly automated smart factories enable more efficient and individual production methods as well as greater customer focus. This includes the comprehensive control and organization of the entire value chain by processing of real-time data from all stages involved in the chain. Cyber-Physical Systems (CPS) [2], which consist of information technology, machines, and built-in sensors, form an unit that enables comprehensive optimization of production with regard to criteria such as costs, resource consumption, quality, or availability. There is often a strong focus on autonomous systems and the highest possible degree of automation. Yet in highly complex processes, human involvement remains indispensable. Often the production process depends on activities in which people intervene, perform complex activities and make important decisions.

Such business and production processes are typically governed by Business Process Management Systems (BPMS) [3] also known as Process-Aware Information Systems (PAIS). BPMS are in charge of the sequencing of the different activities belonging to a process including automated activities and those processed by human agents. The success of any BPM process realization can be endangered by excessive activity automation and poor design of work assignment strategies [4]. Therefore, assigning the optimal agent to an activity and vice versa is a time consuming but necessary task with every BPMS. In most BPMS, so called Staff Assignment Rules (SARs) are utilized to achieve this. However, in Industry 4.0 production scenarios, many different factors have to be taken into account to find an agent that can process an activity in an efficient and effective manner. An obvious example for such factors is the qualification of the agent: She must have the necessary skills and abilities to correctly execute the activity, yet not be overqualified and this incur unnecessary cost overhead, usually agents with a much higher qualification level should not be assigned to a particular activity. Such optimizations should also consider balancing the agent workload to not overburden an agent while others are idle.

In large production facilities, the physical location of the agents and where the activities are to be performed also play an important role. An example are maintenance activities that have to be executed from time to time across a large number of production machines at a large facility. If not optimized properly, agents may waste a substantial amount of time in transit to activities, analogous to the well-known Traveling Salesman Problem [5]. Due to the high complexity of smart factories and their CPS, involving specialized external workers with specific knowledge to maintain a system can incur additional costs. To contain these costs, utilization of internal employees should be preferred if possible, depending on the urgency, availability, and qualification levels. In modern production, worker safety is also an important factor that is usually regulated by respective laws. that address hazards such as chemical, electrical, heat, and noise and may not be adequately tracked by automation systems.

Taking such factors into account, it becomes evident that standard BPMS SARs are insufficient because they are only capable of determining if an agent is able to perform an activity but cannot determine the degree of suitability. Fuzzy logic's [6] fine granular classification between 0 and 1 provides a way to overcome the limitations of simple Boolean logic and determine a specific assignment score for each agent for each possible assignment. Automating such a generic and recurring activity can optimize work efficiency and manpower cost, while reducing employee frustration when automated systems seem inflexible or make unsuitable assignments.

In prior work [7][8][9], we also developed an approach for contextual process management. However, this approach was tailored towards software engineering processes and did not use fuzzy sets or involve the complex specifics of Industry 4.0 nor AR processes. The main focus of this approach was extending processes with properties to enable automated soft-

 ISBN: 978-1-61208-882-2

ware quality assurance and support collaboration of software engineers. This was realized by automatic process adaptations.

In this paper, we contribute an approach for activity assignment in Industry 4.0 projects that takes the aforementioned factors into account. By applying fuzzy sets, fine grained levels of suitability are integrated to improve resource assignment results. To demonstrate its feasibility, we integrated our solution in a commons BPMS.

The remainder of this paper is structured as follows: Section II highlights related research and background information. Section III then describes the general concept and an initial solution approach, while Section IV details the concept for our IAC. In Section V, we provide specific implementation details focusing on the Intelligent Assignment Component while addressing the overall prototype. Next, Section VI we evaluate our solution. Finally, Section VII provides a conclusion and outlook on future upcoming work.

II. Related Work

In literature, there are numerous approaches for activity assignment optimization, utilizing different algorithms like fuzzy sets. The approach presented by Shahhosseini and Sebt [10] proposes a fuzzy adaptive model for competency-based employee selection. It provides a large set of competencies and a complicated fuzzy model. It is, however, created for construction companies and centers around four specific human roles. It also lacks an integration strategy with BPMS. Similarly, Kłosowski et al. [11] also discuss a fuzzy model for assigning workers to production activities. The main focus of this approach is employee assessment and a rich set of properties. However, for our use case the model is too generic and contains unnecessary properties, while at the same time neglecting other important factors like worker safety or location. Furthermore, it lacks BPMS integration concepts.

Tasdemir and Toklu [12] provide an approach for fuzzy activity assignment integration with BPM concepts. The described system is not suitable for the Industry 4.0 scenario as it focuses on teams and the social relationships of the worker in the team. In addition, some criteria for evaluating the performance of workers are not compatible with current legal regulations. Reijers et al. [13] also proposes a BPM-integrated model for activity assignment, in this case based on swarm intelligence. However, that model is targeted at emergency situations where timeliness is more important than other selection criteria and is thus also not suitable for Industry 4.0 production. In contrast to this,Antonelli and Bruno [14] deals with an Industry 4.0 topic: activity assignment in human robot collaboration. This approach splits the activity assignment problem into activity classification with a decision tree classifier and activity assignment with a decision-making algorithm. However, the approach does not address BPMS integration and relies on Boolean rather than fuzzy values, which makes it somewhat synthetic. In addition, worker safety is not taken into account.

Another approach for activity-resource assignment that applies fuzzy logic is presented by Xu et al. [15]. It contains a comprehensive but complicated fuzzy model targeted at collaborative logistics networks comprising logistics service integrators, activity contractors, and resource providers. Thus, the model cannot be used for the assignment of single workers in Industry 4.0 production. Kwak et al. [16] also present a fuzzy approach for a specific domain, in this case audit staffing for accounting. As with the other approaches, these concepts cannot be easily transported to industrial production scenarios, and they do not address BPMS integration. Finally, a category of approaches similar to Simpson and Roberts [17] utilize various algorithms like Bayesian methods, heuristic algorithms or game theoretic approaches for activity assignment in spatial crowdsourcing. As this domain has rather specific properties on which the algorithms rely, they also cannot readily be applied to Industry 4.0 production and for similar reasons, BPMS integration is not a part of these approaches.

III. Solution Approach

While different approaches for fuzzy activity assignment exist, they are often rather generic and complicated or very specific and tailored to a certain domain. Furthermore, they are mostly not integrated with contemporary BPMS. To overcome these limitations and be able to create a usable system for Industry 4.0 scenarios, we focus on a more concrete model and a specific component executing the activity assignments while addressing integration with current BPMS.

To achieve suitable assignments for this complicated domain in a practical and applicable manner, or approach has to satisfy these requirements.

1) The system shall calculate an assignment score that reflects the suitability level of agents for handling a specific activity.
2) The runtime shall be capable of handling a large number of concurrent assignment scoring requests efficiently.
3) Integration into BPMS shall be readily feasible.

To maximize the efficiency optimization options and support easy integration into various BPMS, a new system for handling assignments is created. By decoupling the assignment process from the BPMS, a separate component can be implemented solely to the assignment process, permitting better performance optimization without the constraints imposed if one were to internally extend a specific BPMS. Furthermore, this decoupling via generic API supports a generic approach that can support integrations across a much wider range of BPMS. The conceptual architecture of the novel Assignment-Engine providing such functionality can be seen in Figure 1.

The Assignment-Engine uses a layer pattern which is further subdivided into modular components, with each of the layers contributing to the final solution. Via the modular layers, if desired, the Data and Algorithm Layer could be directly integrated into BPMS (potentially reducing performance optimization). Alternatively, only the Assignment Handler or its individual components could be directly integrated in a BPMS (with a reduced set of features). Thus, we hitherto focus on the Assignment-Engine as a holistic solution to fulfill all the aforementioned requirements.

 ISBN: 978-1-61208-882-2

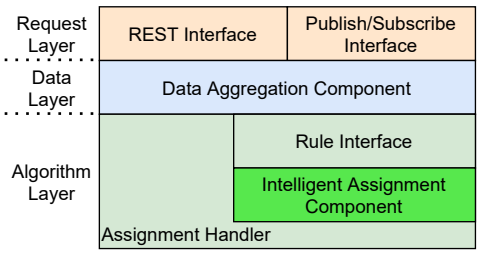

Figure 1. Intelligent-assignment engine conceptual architecture.

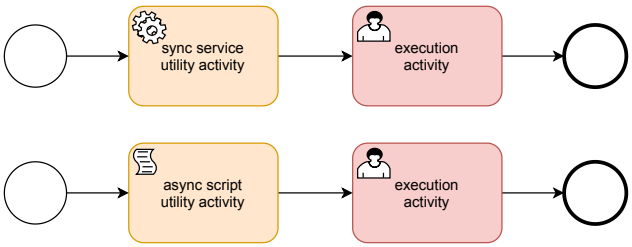

Figure 2. Activity solution variants (synchronous variant on top, asynchronous variant on bottom).

The public REST and Publish/Subscribe (Pub/Sub) Interfaces in the Request Layer are used as a BPMS and programming language independent interface, allowing the usage of the Assignment-Engine with any BPMS supporting BPMN 2.0 or later, as this standard offers a wide range of elements to integrate external services and functions [18] [19]. The integration takes place via a two activity concept. A utility-activity requests an assignment for the execution-activity following in the process workflow. For the utility-activity, two approaches in BPMN 2.0 are possible and should be chosen according to the capabilities of the applicable BPMS (Figure 2).

The Synchronous variant utilizes a Service Task to request the assignment from the Assignment-Engine synchronously. The service activity receives the required data from the process and then awaits the calculated assignment. Finally, it assigns the agent with the highest suitable level to the activity. The Async variant utilizes a Script Task that obtains all the required values itself, accessing the BPMS and then requesting an assignment asynchronously. The BPMS can then ignore the process until the assignment is calculated and no resources have to await a response. As soon as the Assignment-Engine finishes the calculation, it calls the BPMS API and assigns the best fitting agent to the activity itself. With this approach every state-of-the-art BPMS can easily be integrated (requirement 3) and one Assignment-Engine could even support multiple BPMS at the same time.

Once a request is received by the backend, the Data Aggregation Component (DAC) validates it. If all required information is present, the request is sent directly to the Assignment Handler. If some data is still missing the DAC can receive this by predefined external Sources, e.g., a database containing the agents or the BPMS public API.

In order to provide the desired assignment score with a fine granular suitability level, fuzzy sets are chosen. As seen in Section II fuzzy approaches are able to generate very precise assignment scores (requirement 1) in an efficient way (requirement 2). This is an improvement over currently employed chaining-based SARs, which are capable of calculating accurate assignments, but lack the capability to differentiate between suitable agents and therefore do not provide overall optimal assignments. While a Machine Learning (ML) approach could also be a feasible option, the fuzzy sets provide some striking advantages. For fuzzy sets, no preexisting datasets are required, and necessary weights can be configured according to experiences rather than actual data. This enables more traditional companies with weak digitalization and low to no sensor coverage an intelligent assignment capability without a costly and long running initialization phase and can transfer the intelligent assignment with adapted weights instantly to all parts of its production and workflow. Furthermore, our intelligent assignment approach without the training phase typically required by ML approaches.

In order to process the assignment score and assign the best suited worker, the Assignment-Engine either can directly assign the agent via REST-API (as present in many of the most popular BPMS), or the assignment could be conducted in the sync service utility activity via script access to the BPMS from within the process itself.

Due to the complete decoupling of BPM and Assignment-Engine, the latter can be scaled independently of the scaling of the BPMS, and high workload on one of these engines does not interfere with the performance of the other engine. The separation further allows the implementation of an optimized multiprocessing and scaling functionality, guaranteeing optimal efficiency even at high load (requirement 2). The performance optimization takes place at different levels. First a multi-threading approach is utilized in the Request Layer following default architectures for REST and Pub/Sub APIs. The subsequent handling of the request in the Data and Algorithm Layer is handled in a separate process decoupled from the Request Layer. To further speed up large assignment calculations the Intelligent Assignment Component has its own scaling function introduced in Section IV.

Figure 3 shows a simplified workflow graph of a BPMN process consisting of an asynchronous script utility activity requesting an assignment from the Assignment-Engine and the execution activity being assigned after the calculation.

IV. Intelligent Assignment Component

The Intelligent Assignment Component (IAC) is a standalone component of the Assignment-Engine. Containing the fuzzy logic for the assignment calculation it is the functional core of the engine. This section highlights the conceptual decisions behind the component and details the internal structure.

A. Models

In order to compute meaningful assignment scores, the component requires a custom set of models. While Agents and Activity are supplied with each assignment request the

 ISBN: 978-1-61208-882-2

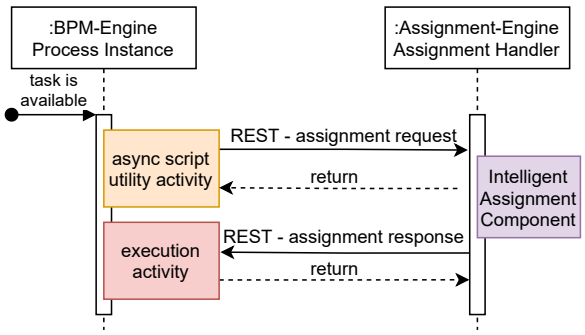

Figure 3. Asynchronous integration in BPM-engine.

Assessment Criteria provides a model which the activity and agent data sets need to include. The agent and activity data can either be directly sent by the BPMS, or be actively collected by the DAC via the BPMS API and connected data sources.

1) Assessment Criteria Model: The Assessment Criteria consist of five parameters, oriented on realword examples and define the values with which the engine should conduct the assignment and which are required in activities and agents used with the IAC. It can be viewed as an interface required by all data and components connected to the assignment.

Distance: calculates the distance between agent and activity position optimizing assignments regarding the travel distance. Position objects contain 3D coordinates with numeric values for X, Y and Z.

Qualification: calculates the difference between the required qualification for an activity and the existing qualification of an agent. It answers the Boolean question if the agent is capable of performing the activity and permits the determination of a possible overqualification to prevent utilizing expensive agents on trivial activities. Qualification objects consist of the four parameters: ”electrical”, ”computer”, ”engineering” and ”bio_chemical”, which represent the different skills of agents or activity requirements in this area. As this skill cannot be calculated automatically and must be defined by humans each parameter will be represented by a number between 0-10. This provides an accustomed scale to rank skill and requirements instead of a default fuzzy scale from 0-1, which is more abstract and an untypical scale for people.

Hourly Rate: calculates the extra cost of using a given agent for an activity per hour in Cents. This prevents the usage of external/temporary workers that incur extra costs if a similar qualified employee is available. This should not include the salary of permanent staff, as their salary is independent of their utilization rate. The cost is represented by an integer to prevent floating errors.

Workload: calculates the capacity utilization of agents, preferring agents with few enqueued activities and preventing overloaded agents form enqueueing additional activities. Thus, load balancing between resources and compliance with labor protection regulations can be supported within the algorithm. This could either be added to the agent itself by the BPMS, or can be calculated using the agents work list, which is present in all BPMS. The parameter is represented by an integer value.

Danger Level: calculates if an agent can safely perform an activity. As some activities have special hazards and given safety regulations, only agents with an appropriate safety clearance can be assigned to these. The Danger is thereby defined by an Object consisting of the four parameters: ”noise”, ”heat”, ”electrical”, and ”chemical”. The separate values are represented by floats between 0 and 1. This provides an abstract concept, but can easily be modified for more concrete parameters as required a concrete use case.

2) Activity Model: This model can consist of any BPM activity extended with the Assessment Criteria except Hourly Rate and Workload, as the cost of an activity is irrelevant for its optimal assignment, and an activity itself has no workload.

3) Agent Model: The model consists of a user from the used BPMS extended with the Assignment Criteria. In a minimal engine such a user could only contain an ID. In contrast to the activity, all criteria are mandatory, as they all provide valuable data for calculating optimal suitability levels. The Danger Level object is renamed to Danger Threshold on the agent level for a more descriptive and easier-to-understand naming. If a task is assigned to an agent the danger level of the activity is subtracted from the agent’s threshold, preventing an overload with too many dangerous activities. After a resting period, the agents danger levels are reset.

B. Overall Assignment Algorithm

The internal algorithms in the IAC are based on a Fuzzy Logic approach. In contrast to ML, no existing datasets are required, only a scheme of the data is mandatory to configure the fuzzy sets. However, the same level of fine calculation of the suitability score is possible as opposed to the simple calculation of suitability based on chaining. As described earlier the activity and agent list are provided to the component either directly by the BPMS or the DAC according to the introduced models and can therefore be directly supplied to the algorithms as parameters without further aggregations or parsing. After executing the algorithms, the IAC will return the suitability level for all provided agents to the Assignment Handler.

To speed up processing time of large numbers of agents the IAC will be able to run calculations in a multi-processing configuration with multiple available modes. This allows an optimal resource allocation concerning the concrete assignments, rather than a general solution that could slow small assignment calculations or non-optimally benefit large calculations.

1) Exclusion Criteria: The Algorithm further contains three exclusion criteria. If one of them is reached, the assignment score for the agent is set to 0 and further calculations are skipped, as the agent is not capable of performing the activity in a qualified or safe way. The agent’s score of 0 will then be returned with the other results at the end of the calculation. The exclusion criteria are:

- Qualification below the required qualification of the activity. This prevents the assignment of unqualified agents to activities even if they are a perfect fit in all other regards.

 ISBN: 978-1-61208-882-2

- Workload equal to or larger than 20, as overloading with activities can lead to stressed out workers, increasing the risk of errors and accidents as well as decreasing their motivation and productivity.
- Danger thresholds lower than the required danger levels of the activity. This not only complies to work safety regulations but also prevents accidents.

2) Fuzzy Calculation: After providing the activity and agent data to the IAC, the calculation can begin. Overall, the agent with the highest score is preferred. Therefore, all Assignment Criteria for each requested agent are calculated distinctly, with 1 being the best score possible and 0 the worst. The concrete calculation for each value is conducted as follows:

Distance: the distance between activity and agent is calculated from the difference between the position variables of both. These are transformed into three dimensional vectors (x, y, z) and deducted. The distances between 0 and 1000 are mapped to a fuzzy value from 1 to 0, where all distances above 1000 are also mapped to 0 in order to reduce the distance agent should travel between activities at maximum. Therefore, agents in close proximity to the activity are preferred in the assignment and all agents further than 1000 units away are heavily discriminated by the fuzzy set. However, as this is no exclusion criteria they could still be assigned to the activity if they are the overall best suited agent.

Workload: the workload can take values between 0 and 20. Values between 0 and 20 are mapped to the fuzzy value between 1 and 0, while values equal to or greater than 10 are mapped to 0. This allows the assignment of new activities to workers with 10 or more activities, but prefers those with lower workloads, Further, if the workload has a value of 20 the score is set to 0 and acts as an exclusion criterion, preventing the algorithms from overburdening agents.

Danger Thresholds: for each danger value, a separate fuzzy set is calculated. After the disjunct calculations, all values are added to a common fuzzy domain and weighted according to a configuration. In the default case, all values are weighted the same, leading to a 25% weight per value. All danger values between 0 and 1 are mapped to fuzzy values from 0 to 1, where all values below the activity's danger level are 0 and trigger an exclusion criterion. This prevents labor law violations and moreover increases work safety. All values above the requirement through the maximum danger threshold of 1 are mapped between 0 and 1. An agent who approximately meets the requirements can therefore work on the activity but gets a score of 0. This prefers agents with higher danger thresholds, as they are most likely more experienced and more rested than agents with lower danger thresholds.

Qualification: the qualification is calculated in three separate fuzzy models. First the four values of qualification between 0 and 10 are compared to the values of required qualification of the activity between 0 and 10 via separate fuzzy sets similar to the danger levels. All values below the activity's requirements are assigned to 0 and trigger an exclusion criterion, as the agent is technically not capable of performing the requested activity. All values above the requirement are assigned to 1. Subsequently, the degree of overqualification is calculated in the second fuzzy model. Starting from the required qualification up to the max qualification of 10, each qualification value is assigned to a fuzzy value between 0 to 1, where 0 perfectly fits the required qualification value and 1 is the maximum amount of overqualification possible. Afterwards the overqualification is subtracted from the qualification value resulting in a value between 0 and 1 called degree of qualification, where 1 is a perfect fit without over qualification and 0 is a maximum overqualification. After this, two steps are conducted for all four properties of an agent's qualification: the four separate degrees of qualification are added to the final fuzzy domain and weighted according to the configuration. The process is identical to the Danger Threshold calculation and also uses a 25% weight distribution per value as a default. The resulting value is used as the qualification in the final calculation of the score, preferring qualified agents with as low an overqualification as possible.

Hourly Rate: the hourly rate is mapped to the fuzzy value from 1 to 0 for values from 0 to 50000 (being equal to 500€ following the integer data format). All values over 50000 are set to 0. This prefers agents with low additional cost like employees over external workers costing extra money and therefore improves the economic efficiency of the BPMS.

After the calculation of the separate suitability levels, if no exclusion criteria is fulfilled, the assignment score is calculated for the agent according to a weight function in one final fuzzy domain. This weight function can either be configured generally and used for all calculations as a default or it can be defined on a per request basis when provided to the Intelligent Assignment Components algorithm as a parameter. All weights are defined between 0 and 1 to a total sum of 1. For the calculation, a new fuzzy set is defined and all values are summed up according to their weight. The result of the new fuzzy set is then multiplied by 100 to return a score between 0 and 100. Finally, the results are sorted in descending order and returned to the Assignment-Engine for further processing.

V. Implementation

Our prototype of the Assignment-Engine with focus on the IAC is implemented using Python. This approach was chosen for its fast prototyping capabilities while still providing performant libraries and refined multiprocessing logic. As a base image for the Assignment-Engine, a Django server was created, providing the most powerful REST-Server available for Python. In contrast to other Python server-frameworks, Django offers not only fast and simple prototyping capabilities, but can also be scaled up to a performant production deployment. To provide the required REST interface, the Django REST framework was integrated. A Pub/Sub interface was implemented using Python paho, the Python MQTT [20] framework from Eclipse. The DAC was added according to the architecture right after REST Layer. It is able to invoke REST requests on its own, aggregating all required data from the BPMS or configured external data sources.

 ISBN: 978-1-61208-882-2

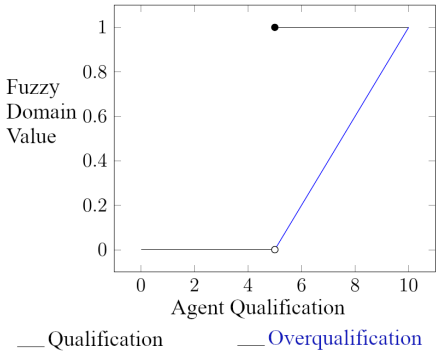

Figure 4. Fuzzy model showing fuzzy domain value vs. agent task qualification.

The fuzzy portions of the IAC were implemented using the fuzzylogic library for Python 3 [21]. As an example, Figure 4 shows the qualification and overqualification fuzzy model for the qualification values as introduced in Section IV. In this example, a qualification of 5 is required for the task. The black line represents the qualification while the blue line represents the overqualification. Below 5 the qualification value is 0, at 5 and above the qualification jumps to 1 and the overqualification starts to rise with a value of 0 by 5 and a maximum of 1 by 10. These two domains are calculated for all four qualification values and afterwards combined in a single domain. The code calculating these fuzzy domains can be seen in Figure 5.

As a BPMS for our prototype, Camunda [22] was chosen. It is a well-known application in the BPM context and further provides all required functionality as well as a BPMN Modeler as an open-source solution. In addition to a full implementation of the BPMN 2.0 standard, Camunda also provides a Connector element, allowing easy REST requests from within process instances via script and service activities.

As the free version of Camunda only provides a BPMS with minimal user management an extension in form of a minimal REST-Backend (further called CamundaClient) handling users and assignments was required. Users are added via a new backend and saved according to our agent model. The process templates were extended as planned in the solution approach. The utility activity requests a score calculation from the CamundaClient for the following execution activity. This execution activity must contain the Assignment Criteria as described in the activity model. The CamundaClient then loads the required user data from the database and sends a request to the Assignment-Engine. It is also possible to move this step to the DAC in the Assignment-Engine, in this case it would only be required to send the activity ID to the Assignment-Engine. While we implemented both, the sync and async variants, we focus on the asynchronous one as it provides more benefits, such as better multiprocessing support, and should be chosen if supported by the utilized BPMS. As soon as the assignment is calculated, the assignment scores are sent from the Assignment-Engine to the CamundaClient and the assignment in Camunda is handled via the client. Connected

```python
def eval_qualification(required: dict, values: dict):
    """eval qualification and over qualification for tasks
    required value and agents value"""
    qualifications = {"types": {}}

    for qualification_type, required_qualification in required.items():
        Qualification = Domain("qualification", 0, 10, res=1)
        qualified = (
            rectangular(required_qualification, 10)
            if required_qualification < 10
            else singleton(required_qualification)
        )
        over_qualified = (
            Set(R(required_qualification, 10))
            if required_qualification < 10
            else Set(constant(0))
        )
        begin_flat = Set(singleton(required))

        if required_qualification != 0:
            begin_flat = MAX(
                ~Set(rectangular(0, required_qualification)),
                Set(singleton(required_qualification)),
                )

        Qualification.qualified = qualified
        Qualification.over_qualified = over_qualified
        Qualification.not_over_qualified = (
            Set(S(required_qualification, 10))
            if required_qualification < 10
            else Set(begin_flat)
        )

        if required_qualification != 0 and required_qualification != 10:
            Qualification.not_over_qualified = product(
                begin_flat, Set(S(required_qualification, 10))
                )
        if required_qualification == 10:
            Qualification.not_over_qualified = begin_flat

        qualifications["types"][qualification_type] = float(
            Qualification.min(values[qualification_type])
        )

    Qualification = Domain("qualification", 0, 1, res=0.001)
    weight = {"electric": 0.25, "computer": 0.25, "social": 0.25, "bio_chemical": 0.25}

    w_func = weighted_sum(weights=weight, target_d=Qualification)
    qualifications["weighted"] = w_func(qualifications["types"])

    return qualifications
```

Figure 5. Code snippet showing fuzzy implementation for qualification.

to this, the workload of the assigned agent(s) is increased and their Danger Threshold is decreased by the Danger Level of the Activity. The Danger Levels further can be reset to the agents' default value, e.g., on daily or weekly base as required by labor safety laws.

Alternatively, the IAC could be integrated in the CamundaClient itself, removing the need for the additional REST-Requests between the client and the Assignment-Engine. The current approach however, allows the usage of the Assignment-Engine as a service for multiple BPMS simultaneously in a generic way.

Powerful multiprocessing capabilities were implemented in the Intelligent Assignment Component and managed by an

 ISBN: 978-1-61208-882-2

intelligent orchestrator. While the Assignment Component is already realized with a runtime of $O(n)$, its performance can be further increased with our multiprocessing approach. Assignments with large numbers of agents can therefore be run in a multi-processing configuration with multiple modes. The default for large requests is $n-1$ processes, where n is the maximal number of cores available on the machine. This provides maximum calculation speed while still preserving one process for the Assignment-Engine itself, preventing slowdowns. If the request is too small for multiprocessing (the multiprocessing overhead would slow down the computation speed), the orchestrator runs the calculation in a single process. Finally, it is possible to run the calculation in $n-m$ processes, where $m, m < n$ is calculated according to the server's performance in multiprocessing mode. We implemented a semi-automatic test setup, calculating the optimal m for a server for 10, 100. 1000, ... 1000000 agents in a single assignment request. The calculated m can then be used in the server configuration to allow maximum performance according to the utilized hardware.

VI. Evaluation

The evaluation is separated in two parts. First the IAC is evaluated on their own regarding its speed, this part is further referred to as the performance evaluation. In the second part, an integration test is conducted comparing the activity assignment of a BPMS (Camunda) with the activity assignment of the Intelligent Assignment Component, referred to as the integration evaluation.

A. Performance Evaluation

The first test was conducted on a virtual server with 90GB main memory. As an operating system Debian 10 was chosen utilizing Python 3.7.2 for the algorithm execution. The test was separated in two groups of 10, 100, 1000, 10000 and 100000 agents getting assigned to a single activity. In the first group all agents were capable to perform the activity according to the assignment criteria. In the second group, only certain agents were capable of performing the activity. The assignment of each group of agents to their activity was conducted 100000 times. The groups of 10 - 1000 agents were assigned using the IAC without multiprocessing while 10000 and 100000 agents were assigned using 17 processes ($n-1$ mode).

Figure 6 displays the assignment calculation performance if all supplied agents were capable, while Figure 7 displays the calculation performance if only some agents fulfilled the requirements. The calculation duration results show an approximately linear scaling in the single processing mode (10-1000 agents), while multiprocessing decreases with larger numbers of agents (10000-100000 agents). Unexpectedly, calculation duration for assignments with only capable agents is lower than that of agents with mixed requirements. This could mean that there are still some optimization problems in the elimination of incapable agents.

In general, the assignment of high volumes of agents provided no problems for the algorithms. As the IAC is meant

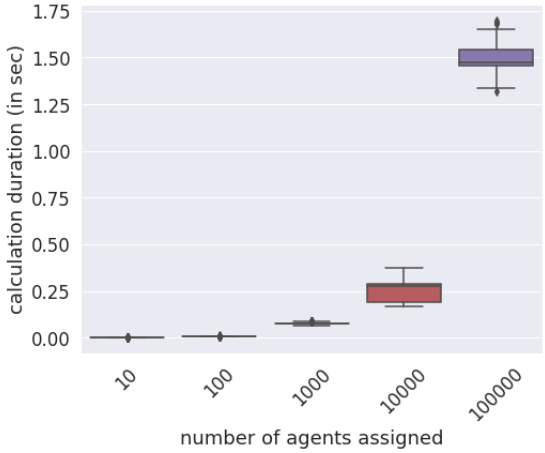

Figure 6. Calculation performance vs. number of capable agents assigned.

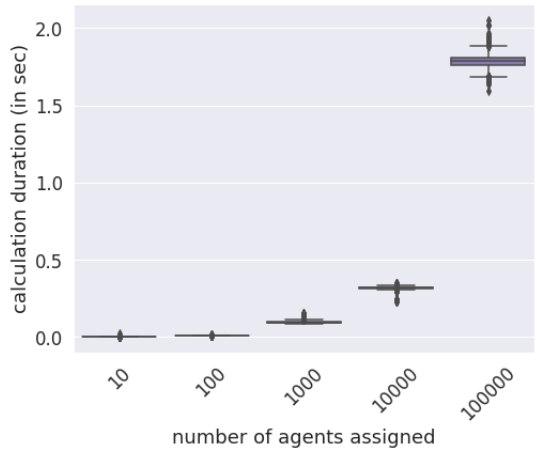

Figure 7. Calculation performance vs. number of capable and non-capable agents assigned.

to run behind SARs, rule engines or other performant basic filtering algorithms a load of 10000 possible agents for a single activity is further quite unlikely. The runtime in the sub seconds for agent values below 10000 would also allow the removal of preliminary filtering, reducing the runtime of the whole BPM process.

B. Integration Evaluation

The second test was conducted using the AnyLogic simulation software. The AnyLogic simulation was run on a Lenovo T495 with 14GB main memory utilizing Arch Linux as an operating system. No changes were made between the performance evaluation and this one besides the setup of this edition. The laptop and server containing the BPM and Assignment-Engine were on the same network.

The evaluation was used to compare a BPMS using the IAC against a plain BPMS. To simulate workers and a realistic workflow, an AnyLogic simulation was built and two simulation setups were configured.

A factory with $21504m^2$ and a total of 29 machines which required maintenance every 16 hours was created. The first

 ISBN: 978-1-61208-882-2

TABLE I. IA/CMD-SETUP SIMULATION MEASUREMENTS.

	IA	Camunda
total_activities (amount)	13.98	16.84
work_time (in minutes)	1636.38	1955.20
idle_time (in minutes)	523.62	204.80
cost (in €)	10.00	420.00
avg_overqualification (value)	0.34	0.09
max_avg_under-qualification (value)	0.00	-0.02
traveled_dist (in meters)	7346.92	8911.79
downtime_maintain (in minutes)	484.18	303.30
downtime_repair (in minutes)	204.00	138.23

TABLE II. INTERNAL/EXTERNAL WORKER SIMULATION MEASUREMENTS.

	IA-int	IA-ext	CMD-int	CMD-ext
total_activities	17.38	0.40	16.85	16.80
work_time	2037.25	32.88	1946.62	1989.50
idle_time	122.75	2127.12	213.38	170.50
cost	0.00	10.00	0.00	420.00
avg_overqual	0.05	1.50	0.06	0.20
max_avg_uqual	0.00	0.00	-0.02	0.00
traveled_dist	9082.78	403.45	8750.17	9558.25

maintenance was scheduled between 0 to 16 hours after start of the simulation. Further, the machines had an average breakdown interval of 36 hours. If a machine required maintenance or repair it started a new Camunda process instance with the required qualification and the machines position. The activity takes between 1 to 3 hours and requires an engineering qualification of 4 for maintenance and 6 for repairs. Other qualifications (electric, computer, bio_chemical) were not required and set to 0. A total of 5 agents were available to complete this activity. Four internal workers, waiting in a maintenance building in the factory hall and one external agent, waiting 165 meters away. The internal agents had engineering qualifications of 4, 5, 6 and 7 while the external agent had an engineering qualification of 8. The other qualification values were set to 0, to avoid bias. The usage of the external agent further was connected to an additional cost of 2500 (25€/activity), while the usage of internal workers incurred no additional costs. In their idle state, an agent checked every 5 minutes if a new activity is available. If they were working, after completion of their current activity they checked if another activity was enqueued. If no activity was enqueued, they switched back to the idle state and moved to their starting position. This part of the setup was identical in both simulation setups.

In the Camunda Setup (called CMD-Setup), the agents fetched their activities directly from Camunda. All activities of the simulation were available to all of the workers with no further verification. If an activity is available to the group, the agents try to claim it and, if successful, work on it. In the IAC Setup (called IA-Setup) the agents checked their personal worklist at the Assignment-Engines REST API. If their personal worklist contains an activity, they start to work on it, otherwise the stayed idle.

A timespan of 36 working hours were simulated for both configurations, using the same seed for the simulations random number generator. This process was repeated 10 times with different seeds to get the statistical relevant test data. For the IAC, the model introduced in Section IV was used. The qualification value was weighted half to increase utilization of the more qualified agents and reduce the downtime of the machines. Further adjustment of the weighting could lead to heavily deviating results. An optimal weighting has to be configured according to the needs of the activities.

Table I shows a general comparison between the CMD and IA simulation, while II shows a more detailed comparison of internal and external worker stats in both simulations. In the following values from Table I will be compared with the more detailed values from Table II.

The average work time and total activities per worker are lower in the IA run, while the utilization of the internal workers (IA-int) is slightly increased and the external utilization (IA-ext) is heavily reduced. The average idle time is increased which can be deducted from the low external utilization. The heavily reduced average cost of a simulation run, if using the IAC instead of a plain BPMS can be attributed to the preferred use of internal workers.

The increase in overqualification while using IA instead of plain Camunda can be explained with the low weighting of qualification in the algorithms as well as no presence of under-qualification in comparison to the CMD-Setup, where under-qualification was generally present. In Table II, the main source of overqualification in the IA simulation comes from the usage of the external worker, who was mainly used for activities below his qualification. This happened because of a too high workload and could be solved by employing another internal worker with lower qualification to help out with this activity. This would lead to reduced cost and downtime. Optimization in the simulated company is needed, rather than an adaptation of the algorithm.

The traveled distance for the internal workers is slightly increased in the IA simulation compared to the CMD run. This however stands in linear dependency with the increased workload. A stronger weight regarding the distance could reduce this effect.

The downtime in the IA run is around 50% higher than in the CMD-Setup while the cost was reduced to 4.2% of the CMD-Setup. This was expected behavior as the algorithms by default try to save money and therefore did not employ the external worker as much as the CMD-setup.

In summary, the IAC worked as expected with fast runtimes on mid-to-low budget hardware. Scaling was only required for the case when more than 1000 agents could be assigned to the same activity. This is highly unlikely even in companies with more than 1000 employees, as many of them would most likely not fulfill the preconditions to be considered for the activity. However, if scaling is necessary, it can be readily achieved and works efficiently for at least 100000 agents per activity. The algorithms further produce comprehensible results for analysis by non-experts, which can be adjusted as required through dynamic weighting of the different variables in the algorithm.

 ISBN: 978-1-61208-882-2

VII. Conclusion and Further Work

Industry 4.0 stands for highly automated production processes. However, these processes also rely on complicated tasks that can only be performed by humans manually. The integration of such activities into the processes is still problematic. One important issue is efficient task assignment, which is not solved well in contemporary BPM systems.

To counteract this, this paper described an approach for more effective and efficient activity assignment for Industry 4.0 production processes. The focus of this approach was to build a compact model of fuzzy sets that can be easily applied to real projects. Therefore, we chose a set of important properties that incorporate aspects relevant in current Industry 4.0 production: achieve cost savings by incorporating not only underqualification but also overqualification and the separation between internal and expensive external workers; achieve a balanced workload for all workers to avoid idle times as well as overburdened workers; protect the workers from different hazards as enforced by legal regulations; and finally, optimize assignments with knowledge about the locations of workers and their potential activities by minimizing transit overhead.

Besides providing a practical model, our approach also features concepts for the direct integration with BPMS. To demonstrate its feasibility, we have currently implemented, integrated and tested our prototype approach with two concrete BPMS (AristaFlow [23] and Camunda). The approach is built modularly and can be easily expanded. Further, the fuzzy weights used to calculate the assignment can be changed dynamically according to the users' specific needs. It is also possible to use this prototype with any other BPMS supporting BPMN 2.0 with minimal effort.

The evaluation showed that our approach is an efficient way to automatically compute assignments. We evaluated the algorithms regarding performance and built a comprehensive simulation scenario to show its effectiveness and efficiency in providing optimal assignment recommendations.

As future work, we plan to incorporate a more generic model where not only the weights are dynamic but also the criteria. Thus, the approach can be easily adapted for other domains and scenarios by extending or replacing the evaluation criteria. Other upcoming improvements could include utilizing transit path finding algorithms for the distance calculation to provide a more realistic and resilient calculation. The duration of activities could also be considered in order to measure the workload not only in terms of number but also in terms of estimated time required to complete the activities. Finally, the slight performance decrease in the elimination of unqualified agents should be investigated.

Acknowledgments

We thank Felix Gräber for his contribution regarding the fuzzy algorithms and implementation, and Camil Pogolski for his input on the models. This work was partially funded via the PARADIGMA project by "Zentrales Innovationsprogramm Mittelstand" (i.e., the "Central Innovation Programme for small and medium-sized enterprises (SMEs)"), of the Federal Ministry for Economic Affairs and Energy of the Federal Republic of Germany.

References

[1] H. Lasi, P. Fettke, H.-G. Kemper, T. Feld, and M. Hoffmann, "Industry 4.0," *Business & Information Systems Engineering*, vol. 6, no. 4, pp. 239–242, 08 2014.

[2] R. Baheti and H. Gill, "Cyber-physical systems," *The impact of control technology*, vol. 12, no. 1, pp. 161–166, 2011.

[3] D. Karagiannis, "Bpms: business process management systems," *ACM SIGOIS Bulletin*, vol. 16, no. 1, pp. 10–13, 1995.

[4] C. Moore, "Common mistakes in workflow implementations," *Giga Information Group, Cambridge, MA*, vol. 2, 2002.

[5] M. M. Flood, "The traveling-salesman problem," *Operations Research*, vol. 4, no. 1, pp. 61–75, Feb. 1956.

[6] L. A. Zadeh, "Fuzzy logic," *Computer*, vol. 21, no. 4, pp. 83–93, 1988.

[7] G. Grambow, R. Oberhauser, and M. Reichert, "Semantic workflow adaption in support of workflow diversity," in *4th Int'l Conf. on Advances in Semantic Processing*. Xpert Publishing Services, 2010, pp. 158–165.

[8] G. Grambow, R. Oberhauser, and M. Reichert, "Employing semantically driven adaptation for amalgamatingsoftware quality assurance with process management," in *Second Int'l Conf. on Adaptive and Self-adaptive Systems and Applications (ADAPTIVE'10)*. Xpert Publishing Services, 2010, pp. 58–67.

[9] G. Grambow, R. Oberhauser, and M. Reichert, "Enabling automatic process-aware collaboration support in software engineering projects," in *Communications in Computer and Information Science (CCIS) 303*. Springer, 2012, pp. 73–89.

[10] V. Shahhosseini and M. Sebt, "Competency-based selection and assignment of human resources to construction projects," *Scientia Iranica*, vol. 18, no. 2, pp. 163 – 180, 2011.

[11] G. Kłosowski, A. Gola, and A. Świć, "Application of fuzzy logic in assigning workers to production tasks," *Distributed Computing and Artificial Intelligence, 13th International Conference*, pp. 505–513, 01 2016.

[12] C. Tasdemir and C. Toklu, "Qos driven dynamic task assignment for bpm systems using fuzzy logic," in *Emerging trends and challenges in information technology management*. Hershey, Pennsylvania (701 E. Chocolate Avenue, Hershey, Pa., 17033, USA: IGI Global, 2006.

[13] H. A. Reijers, M. H. Jansen-Vullers, M. zur Muehlen, and W. Appl, "Workflow management systems + swarm intelligence = dynamic task assignment for emergency management applications," in *Business Process Management*, G. Alonso, P. Dadam, and M. Rosemann, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2007, pp. 125–140.

[14] D. Antonelli and G. Bruno, "Dynamic distribution of assembly tasks in a collaborative workcell of humans and robots," *FME Transactions*, vol. 47, pp. 723–730, 01 2019.

[15] X. Xu, J. Hao, L. Yu, and Y. Deng, "Fuzzy optimal allocation model for task–resource assignment problem in a collaborative logistics network," *IEEE Transactions on Fuzzy Systems*, vol. 27, no. 5, pp. 1112–1125, 2019.

[16] W. Kwak, Y. Shi, and K. Jung, "Human resource allocation in a cpa firm: A fuzzy set approach," *Review of Quantitative Finance and Accounting*, vol. 20, pp. 277–290, 2003.

[17] E. Simpson and S. Roberts, *Bayesian Methods for Intelligent Task Assignment in Crowdsourcing Systems*. Cham: Springer International Publishing, 2015, pp. 1–32.

[18] *Business Process Model and Notation(BPMN) - Version 2.0.2*. Object Management Group, 12 2013, retrieved: 2021.06.10. [Online]. Available: https://www.omg.org/spec/BPMN/2.0.2/PDF

[19] T. Allweyer, *BPMN 2.0: introduction to the standard for business process modeling*. BoD–Books on Demand, 2016.

[20] G. C. Hillar, *MQTT Essentials-A lightweight IoT protocol*. Packt Publishing Ltd, 2017.

[21] A. Kiefner. Python3 fuzzylogic. Last visited: 2021.06.14. [Online]. Available: https://github.com/amogorkon/fuzzylogic

[22] Camunda. Last visited: 2021.06.10. [Online]. Available: https://camunda.com

[23] M. Reichert, "Enabling flexible and robust business process automation for the agile enterprise," in *The Essence of Software Engineering*. Springer, Cham, 2018, pp. 203–220.

 ISBN: 978-1-61208-882-2

The Architecture of a Software Framework for Biologically-Inspired Optimization Algorithms

Florin Leon
Faculty of Automatic Control and Computer Engineering
"Gheorghe Asachi" Technical University of Iași
Iași, Romania
e-mail: florin.leon@academic.tuiasi.ro

Silvia Curteanu
Faculty of Chemical Engineering and Environmental Protection
"Gheorghe Asachi" Technical University of Iași
Iași, Romania
e-mail: silvia.curteanu@academic.tuiasi.ro

***Abstract*—Biologically-inspired optimization algorithms are robust techniques that can be applied for a wide range of practical problems. A large number of such algorithms have been proposed by researchers, but their implementation is often not available or is done in different programming languages. This paper presents a flexible software architecture of a .NET optimization framework where such methods can be easily incorporated. It includes algorithms that belong to various subfields: from ideas based on human social behavior to ideas based on virus behavior. Using this framework, many modifications and hybridizations for algorithms are also possible.**

Keywords-biologically-inspired optimization algorithms; optimization framework; software architecture; .NET framework.

I. INTRODUCTION

Biologically-inspired optimization algorithms have become popular in the recent years as general, simple, and robust techniques that can be used when other mathematical optimization methods cannot be applied. Among their advantages, which facilitate their application to various fields, such as planning, design, control, classification, clustering, and time series modeling, one can mention the following [1]:

- They do not require the objective function to be continuous and/or differentiable;
- They do not require extensive problem formulation, e.g., in case of traditional methods such as integer programming, geometric programming, as well as branch and bound methods, a special mathematical formulation is required for solving a problem;
- They are not sensitive to the starting point of the search;
- They are more robust in the presence of local optima and are more likely to find the global optimum of a function than gradient-based methods [2][3].

In this work, the architecture of an optimization framework called HAVOC for biologically-inspired optimization will be presented. The abbreviation comes from the title of the project: *From Human to Virus. Optimization Algorithms with Chemical Engineering Applications*. It is one of the few optimization frameworks available for .NET, which incorporates biologically-inspired algorithms that belong to a wide range of subfields: from ideas based on human social behavior to ideas based on virus behavior. While many such algorithms have been proposed and some have their implementation publicly available, they are not integrated into a unified framework.

The motivation of dealing with algorithms inspired by human and virus behavior can be seen from several perspectives:

- A relatively new, little studied field of interest is approached, which creates good scientific perspectives in the sense of identifying directions that can contribute to stimulating creativity and innovative ideas;
- There are also few applications of these algorithms, so future work will include applying these algorithms in a new field, i.e., chemical engineering (more precisely, polymerization engineering), with processes that represent a challenge through the complexity and difficulty of the problems they raise in the actions of modeling and optimization.
- A systematic, detailed study of these algorithms is intended, from basic variants to modified versions, from general to the particular, adapting to specific applications and including comparisons with previous achievements, but also with approaches from the literature.

The rest of this paper is organized as follows. In Section II, some related work in terms of other optimization libraries available today is addressed. Section III describes the architecture of the HAVOC optimization framework. The actual algorithms are briefly presented in Section IV. A discussion of future work and the conclusions are included in Section V.

II. RELATED WORK

Presently, there are a number of evolutionary/genetic algorithm libraries available, such as: *Evolving Objects: an Evolutionary Computation Framework* [4][5], *Genetic Algorithms Framework* [6], *Watchmaker Framework for Evolutionary Computation* [7], *AForge.NET Genetic Algorithms Library* [8], as well as many others.

The *SciPy optimize* [9] package contains strategies such

 ISBN: 978-1-61208-882-2

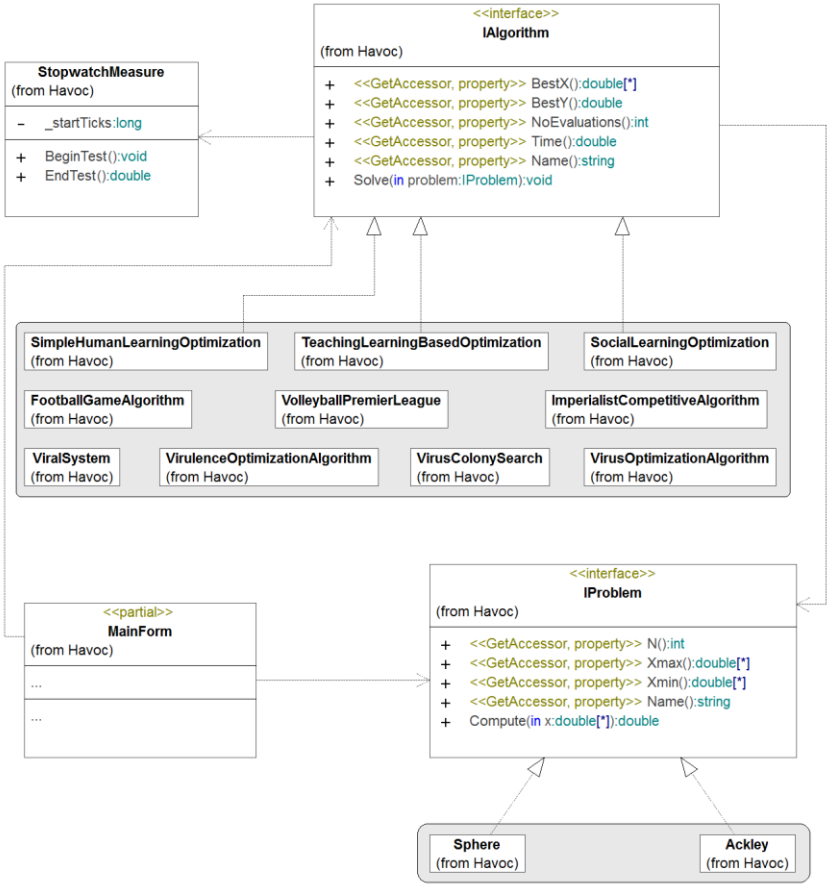

Figure 1. The architecture of the HAVOC framework (UML class diagram).

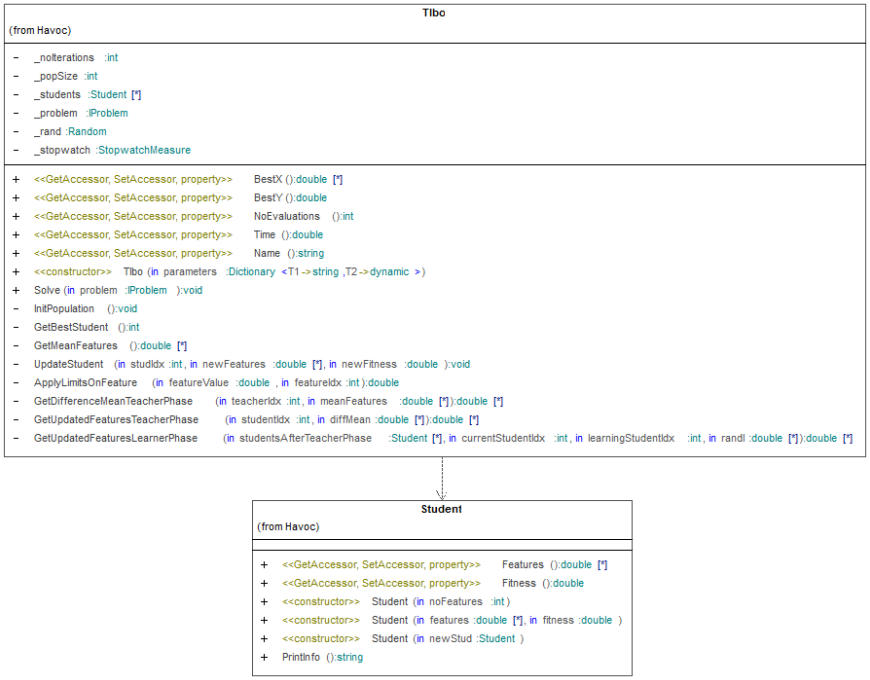

Figure 2. The implementation of the *Teaching-Learning based Optimization* algorithm (UML class diagram).

as Nelder-Mead, Powell, Quasi-Newton strategies, etc., and can also handle linear and nonlinear constraints. *lpsolve* [10] is another powerful mixed integer linear programming solver. *MathWorks* also provides an *Optimization Toolbox* [11] for Matlab, which can solve linear, quadratic, conic, integer, and nonlinear optimization problems. A general evolutionary algorithm framework especially designed for multi-agent systems is presented in [1].

Other optimization systems that can be mentioned are: ECJ [12] and jMetal [13], programmed in Java, DEAP [14], implemented in Python, and HeuristicLab [15]. From these tools, only the last one uses .NET. It also allows the dynamic inclusion of additional algorithms as plug-ins. However, these frameworks do not contain yet the classes of algorithms envisaged to be included in the proposed HAVOC system.

III. SYSTEM ARCHITECTURE

The architecture of the HAVOC framework is presented in Figure 1 as a *Unified Modeling Language* (UML) class diagram. The purpose of the framework is to allow the user to experiment with different algorithms and problems in a very simple way. The user selects a problem and an algorithm, and the framework solves the problem using the specified algorithm. In order to be able to handle this, all ten algorithm classes implement the common *IAlgorithm* interface. Similarly, all the problem classes implement the *IProblem* interface. In this way, the corresponding code is common for all combinations of problems and algorithms:

```
IProblem problem = new XProblem();
IAlgorithm algorithm = new YAlgorithm(parameters);
algorithm.Solve(problem);
```

Afterwards, the solution and additional information can be retrieved from the algorithm object. The *BestX* property represents the "*x*" that minimizes *f*(*x*); it is a vector of real numbers. The *BestY* property is a real number that represents the "*y*", where *y* = *f*(*x*).

Other useful characteristics are the number of function evaluations that the algorithm performs. This is one of the recent ways to compare the performance of heuristic optimization algorithms. The actual execution time is also provided, using the *Stopwatch* class from the .NET framework.

Each algorithm object is instantiated with a set of parameters. More specifically, the parameters are given as a *Dictionary* (a generic hash table collection), where the key is the name of the parameter and the value is the corresponding parameter value:

```
var parameters = new Dictionary<string, dynamic> {
   ["NoIterations"] = 1000, ["PopSize"] = 100 };
IAlgorithm alg = new Tlbo(parameters);
alg.Solve(currentProblem);
// alg.BestX, alg.BestY, alg.NoEvaluations, alg.Time are available
```

These values are not restricted to be real numbers, they can have any desired type. The user will be able to modify the parameter values from the graphical user interface or directly by editing a configuration file.

Each algorithm is implemented in it own class (or classes); e.g., in Figure 2, the UML class diagram for the *Teaching-Learning based Optimization* (TLBO) algorithm is presented. Figure 3 shows the results obtained with TLBO for several well-known benchmark problems: *Sphere*, *Ackley*, *Griewank*, and *Rosenbrock* [16]. They were obtained using a computer with a 4-core 2 GHz Intel processor and 8 GB of RAM.

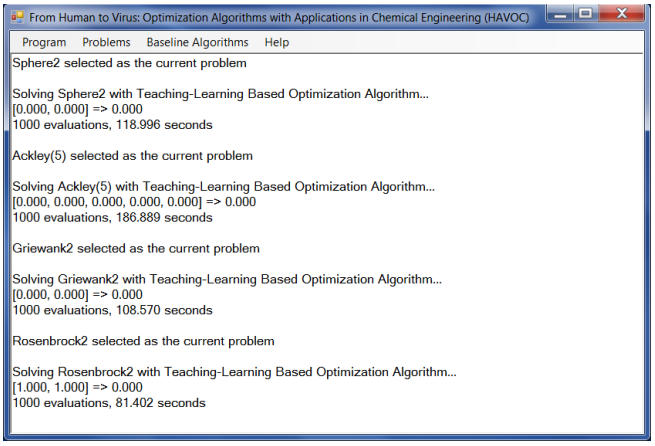

Figure 3. The results obtained with the *Teaching-Learning based Optimization* algorithm for several benchmark problems.

The *IProblem* interface defines the main characteristics of an optimization problem: its dimensionality (the *N* property), the search range defined by the lower bounds and the upper bounds of each dimension (the *XMin* and *XMax* properties), and the actual implementation of the function to be optimized in the *Compute* method. The optimization problems need not be simple, analytical functions. They can be complex, simulation-based procedures, e.g., the minimum distance from a target when launching a space probe with certain initial properties, such as angle, speed and acceleration. The trajectory of this object can be simulated in a discrete, step-by-step fashion, taking into account the resulting force applied by the large space objects on the probe. As long as the simulation returns a value (in this example, the ideal value is 0, i.e., the probe has reached its target), it can be used as an optimization function in the proposed framework.

In the present version, the architecture only supports solutions encoded as an array of real numbers, but solutions represented as an array of integers can also be found. This would require that the user make a genotype-phenotype distinction, where the actual encoding genes are not used as such in the computation of the objective function but they are transformed or interpreted in a certain way. E.g., real numbers can be interpreted as integers, by rounding them or using random key encoding for problems that have permutations as solutions. In future versions, constraints can be included as C# predicates that can be passed as

 ISBN: 978-1-61208-882-2

parameters to the algorithms. Variable length solutions can also be employed using the same array-based representations. However, these changes in the architecture should also be supported by the underlying algorithms.

IV. THE ALGORITHMS

The included algorithms are grouped into three main categories, as shown in Table I. They are discussed in the following subsections.

TABLE I. THE ALGORITHMS SELECTED FOR IMPLEMENTATION

Category	Algorithm	Encoding
Algorithms inspired by the human behaviors of learning and cooperation	*Simplified Human Learning Optimization* (SHLO) [17]	*binary*
	Social Learning Optimization (SLO) [18]	*real*
	Teaching-Learning based Optimization (TLBO) [19]	*real*
Algorithms inspired by human competitive behavior	*Football Game Algorithm* (FGA) [20][21]	*real*
	Volleyball Premier League (VPL) [22]	*real*
	Imperialist Competitive Algorithm (ICA) [23]-[25]	*real*
Algorithms inspired by virus behavior	*Viral System* (VS) [26]	*binary*
	Virulence Optimization Algorithm (VOA1) [27]	*real*
	Virus Colony Search (VCS) [28]-[30]	*real*
	Virus Optimization Algorithm (VOA2) [31]-[33]	*real*

A. Algorithms inspired by the human behaviors of learning and cooperation

Many learning activities are similar to the meta-heuristic search, where the success is achieved through repetition and adjusting. Although the human learning process is very complex, the three main strategies used in human learning include: random learning, individual learning and social learning. SHLO includes all three strategies to search for good solutions. On the other hand, SLO simulates the social learning approach and it is based on the *Culture Algorithm* framework (CA) [34] comprising three elements: population space, belief space and a protocol describing the manner in which knowledge is exchanged between the two spaces. To this CA framework, SLO adds an additional bottom layer that includes individual genetic evolution. In case of the TLBO algorithm, the simulated process is represented by the transfer of knowledge between the teacher and the student, i.e., the teaching phase, and the collaboration between students, i.e., the learning phase. Very few variants and combinations have been developed for the first two algorithms: adaptive SHLO [35][36] and SLO with *Differential Evolution* and improved *Social Cognitive Optimization* [37]. TLBO is the most used algorithm due to its simplicity and efficiency [38], and has been included in various hybrid variants: TLBO with error correction strategy and Cauchy distribution [39], TLBO including genetic crossover and mutation strategies [40], TLBO with *Bird Mating Optimizer* [41], TLBO with *Differential Learning* [42] and more.

B. Algorithms inspired by human competitive behavior

In addition to the concept of learning associated with humans, the concept of competitiveness is studied. FGA mimics the manner in which football is played. The population is represented by the players and the optimization is performed in two steps: random walk and coaching, with two strategies: attacking and substitution. Distinctively from FGA, VPL closely follows the rules of volleyball and the solution is formed from two segments: active (representing the main formulation of each team) and passive (which stores variables and special instruction rules). If FGA and VPL are team-based, in the sense that they include a reduced number of players, ICA simulates the competition behavior at the "national" level, where each individual represents a country and there are two types of countries: imperialist and colonies. For the first two algorithms, very few improvements and applications have been published: modified VPL using sine cosine algorithm [43] and *Multi-Objective Volleyball Premier League* [44]. By comparison, ICA is a more mature algorithm, with many applications, known in different variants, of which one can mention: ICA with different chaotic maps [45], ICA with k-means clustering [46], ICA with neural networks [47][48], etc.

C. Algorithms inspired by virus behavior

VS is based on the viral infection processes and, to reach the optimum, it uses two mechanisms: replication and infection. Depending on the type of virus infection, i.e., selective or massive infection, and on the type of evolution of the virus, different mechanisms are activated during the search. VS stops when the collapse and death of the organism occurs or the virus is isolated. On the other hand, VOA1 simulates several important mechanisms in the virus life-cycle: reproduction and mutation, cloning, and escaping from the infected region. In the case of VCS, the virus diffusion, cell infection and immune response are simulated using Gaussian random walk, *Covariance Matrix Adaptation Evolution Strategy and Evolution Strategy*. VOA2 mainly focuses on converting the concept of a virus attacking a host cell into a continuous domain optimization method. In general, there are few approaches and applications of these algorithms, with a single modified variant [49].

V. CONCLUSIONS AND FUTURE WORK

In this work, the architecture of an optimization framework called HAVOC was presented. It is extensible, so that other algorithms and variants of the base algorithms should be easy to add. In this way, it facilitates comparisons on both benchmark problems and real-life problems from the chemical engineering field. HAVOC uses .NET (C#) and includes classes of algorithms less often implemented in other publicly available frameworks.

Since the open literature presents too few variants and applications of algorithms that mimic the human and virus behaviors, our future research aims to carry out an in-depth, systematic study, accompanied by rigorous tests and various applications, bringing the following as elements of novelty and originality:

 ISBN: 978-1-61208-882-2

- New variants of the optimization algorithms inspired by human, animal and virus behaviors;
- Combinations between different categories of algorithms such as neural networks, support vector machines, evolutionary algorithms, artificial immune systems, etc., leading to new hybrid configurations with improved performance, which can open new perspectives for the advanced modeling and optimization techniques;
- Association of the developed methodologies with new real-world case studies;
- Innovative implementation of the proposed methods by original software.

The new variants of the algorithms, including new concepts, the hybrid configurations of the artificial intelligence tools, and the use of bio-inspired modeling and optimization methodologies for chemical processes can be considered as new approaches, which open new fruitful directions of research and applications in this field. At the same time, the flexibility of the algorithms, the way in which the implementations will be made, their possibilities of adaptation, give them generality and, therefore, possibilities to be applied in various other fields.

Possible applicative research directions derive from the chemical processes considered as case studies for the proposed algorithms. The results of the modeling and optimization procedures provide useful information for experimental and industrial practice, e.g.:

- They can substitute or better schedule the experiments that are materials-, energy- and time-consuming;
- They emphasize the maximum performance of the systems and the conditions necessary to accomplish them;
- These achievements bring important economic benefits, as the use of the developed modeling and optimization techniques represent an important stage in optimal control engineering.

ACKNOWLEDGMENT

This work was supported by Exploratory Research Projects PN-III-P4-ID-PCE-2020-0551, financed by UEFISCDI.

REFERENCES

[1] F. Leon, B. I. Aignătoaiei, and A. D. Leca, “An Evolutionary Optimization Framework for Intelligent Agents”, Proceedings of the 15th International Conference on System Theory, Control and Computing (Joint Conference SINTES 15, SACCS 11, SIMSIS 15), 306-311, 2011.

[2] B. V. Babu and R. Angira, “Modified differential evolution (MDE) for optimization of non-linear chemical processes”, Comp. Chem. Eng., vol. 30, pp. 989–1002, 2006.

[3] S. Curteanu and F. Leon, “Optimization Strategy Based on Genetic Algorithms and Neural Networks Applied to a Polymerization Process”, International Journal of Quantum Chemistry, vol. 108, pp. 617–630, Wiley Periodicals, USA, 2008.

[4] M. Keijzer, J. J. Merelo, G. Romero, and M. Schoenauer, “Evolving objects: A general purpose evolutionary computation library”, Artificial Evolution, vol. 2310, pp. 829–888, 2002.

[5] Evolving Objects (EO): an Evolutionary Computation Framework, http://eodev.sourceforge.net, 2012 [retrieved: June, 2021].

[6] Genetic Algorithms Framework, http://sourceforge.net/projects/ga-fwork, 2013 [retrieved: June, 2021].

[7] Watchmaker Framework for Evolutionary Computation, http://watchmaker.uncommons.org, 2010 [retrieved: June, 2021].

[8] AForge.NET, “Genetic Algorithms Library”, http://www.aforgenet.com/framework/features/genetic_algorithms.html, 2013 [retrieved: June, 2021].

[9] The SciPy community, “Optimization (scipy.optimize)”, https://docs.scipy.org/doc/scipy/reference/tutorial/optimize.html, 2021 [retrieved: June, 2021].

[10] M. Berkelaar, K. Eikland, and P. Notebaert, “lpsolve, Mixed Integer Linear Programming (MILP) solver”, https://sourceforge.net/projects/lpsolve, 2021 [retrieved: June, 2021].

[11] MathWorks, “Optimization Toolbox”, https://www.mathworks.com/products/optimization.html, 2021 [retrieved: June, 2021].

[12] S. Luke et al., “ECJ 27, A Java-based Evolutionary Computation Research System”, https://cs.gmu.edu/~eclab/projects/ecj, 2019 [retrieved: June, 2021].

[13] A. J. Nebro, “jMetal, Metaheuristic Algorithms in Java”, http://jmetal.github.io/jMetal, 2020 [retrieved: June, 2021].

[14] F. A. Fortin, F. M. De Rainville, M. A. Gardner, M. Parizeau, and C. Gagné, “DEAP: Evolutionary Algorithms Made Easy”, Journal of Machine Learning Research, vol. 13, pp. 2171–2175, 2012.

[15] S. Wagner et al., “HeuristicLab, A Paradigm-Independent and Extensible Environment for Heuristic Optimization”, https://dev.heuristiclab.com/trac.fcgi/wiki, 2019 [retrieved: June, 2021].

[16] R. Oldenhuis, “Test functions for global optimization algorithms”, https://github.com/rodyo/FEX-testfunctions/releases/tag/v1.5, 2021 [retrieved: June, 2021].

[17] L. Wang, H. Ni, R. Yang, M. Fei, and W. A. Ye, “Simple Human Learning Optimization Algorithm”, Communications in Computer and Information Science book series CCIS, Springer Berlin, vol. 462, pp. 56–65, 2014.

[18] Z. Z. Liu, D. H. Chu, C. Song, X. Xue, and B. Y. Lu, “Social learning optimization (SLO) algorithm paradigm and its application in QoS-aware cloud service composition”, Information Sciences, vol. 326, pp. 315–333, DOI: 10.1016/j.ins.2015.08.004, 2016.

[19] R. V. Rao, V. J. Savsani, and D. P. Vakharia, “Teaching–learning-based optimization: A novel method for constrained mechanical design optimization problems”, Computer-Aided Design, vol. 43, pp. 303–315, 2011.

[20] E. Fadakar and M. Ebrahimi, “A new metaheuristic football game inspired algorithm”, 1st Conference on Swarm Intelligence and Evolutionary Computation, CSIEC 2016 Proceedings, 6-11, 2016.

[21] A. V. Djunaidi and C. P. Juwono, “Football game algorithm implementation on the capacitated vehicle routing problems”,

 ISBN: 978-1-61208-882-2

International Journal of Computing Algorithm, vol. 7.1, pp. 45–53, 2018.

[22] R. Moghdani and K. Salimifard, "Volleyball Premier League Algorithm", Applied Soft Computing, vol. 64, pp. 161–185, 2018.

[23] E. Atashpaz-Gargari and C. Lucas, "Imperialist competitive algorithm: an algorithm for optimization inspired by imperialistic competition", Evolutionary computation, CEC IEEE Congress, 4661-4667, 2007.

[24] H. Shabani, B. Vahidi, and M. Ebrahimpour, "A robust PID controller based on imperialist competitive algorithm for load-frequency control of power systems", ISA Transactions, vol. 52, no. 1, pp. 88–95, 2013.

[25] S. Hosseni and A. Khaled, "A survey on the imperialist competitive algorithm metaheuristics: implementation in Engineering domain and directions for future research", Applied Soft Computing, vol. 24, pp. 1078–1094, 2014.

[26] P. Cortés, J. M. García, J. Muñuzuri, and L. Onieva, "Viral systems: A new bio-inspired optimisation approach", Computers & Operations Research, vol. 35, pp. 2840–2860, 2008.

[27] M. Jaderyan and H. Khotanlou, "Virulence Optimization Algorithm", Applied Soft Computing, vol. 43, pp. 596–618, 2016.

[28] M. D. Li, H. Zhao, X. W. Weng, and T. Han, "A novel nature-inspired algorithm for optimization: Virus colony search", Advances in Engineering Software, vol. 92, pp. 65–88, 2016.

[29] S. J. D. Hosseini, M. Moradian, H. Shahinzadeh, and S. Ahmadi, "Optimal Placement of Distributed Generators with Regard to Reliability Assessment using Virus Colony Search Algorithm", International Journal of Renewable Energy Research, vol. 8, 2018.

[30] H. Shahinzadeh, G. B. Gharehpetian, M. Moazzami, J. Moradi, and S. H. Hosseinia, "Unit commitment in smart grids with wind farms using virus colony search algorithm and considering adopted bidding strategy", Proceedings of Smart Grid Conference (SGC), 1-9, 2017.

[31] Y. C. Liang and J. R. Cuevas Juarez, "Multilevel image thresholding using relative entropy and virus optimization algorithm", IEEE Congress on Evolutionary Computation, 2012.

[32] Y. C. Liang and J. R. Cuevas Juarez, "A novel metaheuristic for continuous optimization problems: Virus optimization algorithm", Engineering Optimization, vol. 48, pp. 73–93, 2015.

[33] Y. C. Liang and J. R. Cuevas Juarez, "Harmony search and virus optimization algorithm for multi-objective combined economic energy dispatching problems", IEEE Congress on Evolutionary Computation, 3947-3954, 2016.

[34] B. Peng, "Knowledge and population swarms in cultural algorithms for dynamic environments", Wayne State University, 2005.

[35] L. Wang et al., "An adaptive simplified human learning optimization algorithm", Information Sciences, vol. 320, pp. 126–139, 2015.

[36] J. Cao, Z. Yan, X. Xu, G. He, and S. Huang, "Optimal power flow calculation in AC/DC hybrid power system based on adaptive simplified human learning optimization algorithm", Journal of Modern Power Systems and Clean Energy, vol. 4, pp. 690–701, 2016.

[37] A. Naik, A., and S. C. Satapathy, "A comparative study of social group optimization with a few recent optimization algorithms", Complex Intell. Syst. vol. 7, pp. 249–295, DOI: 10.1007/s40747-020-00189-6, 2021.

[38] R. Venkata Rao, "Review of applications of TLBO algorithm and a tutorial for beginners to solve the unconstrained and constrained optimization problems", Decision Science Letter, vol. 5, pp. 1–30, 2016.

[39] Z. Zhai, G. Jia, and K. Wang, "A novel Teaching-Learning-Based Optimization with Error Correction and Cauchy Distribution for Path Planning of Unmanned Air Vehicle", Computational Intelligence and Neuroscience, pp. 1–12, DOI: 10.1155/2018/5671709, 2018.

[40] Y. Kumar, N. Dahiya, S. Malik, and S. Khatri, "A new variant of teaching learning based optimization algorithm for global optimization problems", Informatica, vol. 43, no. 1, 2019.

[41] Q. Zhang, G. Yu, and H. Song, "A hybrid bird mating optimizer algorithm with teaching-learning-based optimization for global numerical optimization", Statistics, Optimization & Information Computing, vol. 3, pp. 54–65, 2015.

[42] F. Zou, L. Wang, D. Chen, and X. Hei, "An Improved Teaching-Learning-Based Optimization with Differential Learning and Its Application", Mathematical Problems in Engineering, pp. 1–19, 2015.

[43] R. Moghdani, M. Abd Elaziz, D. Mohammadi, and N. Neggaz, "An improved volleyball premier league algorithm based on sine cosine algorithm for global optimization problem", Engineering with Computers, pp. 1–30, 2020.

[44] R. Moghdani, K. Salimifard, E. Demir, and A. Benyetton, "Multi-objective volleyball premier league algorithm", Knowledge-Based Systems, vol. 196, no. 21, 2020.

[45] S. Talatahari, B. Farahmand Azar, R. Sheikholeslami, and A. H. Gandomi, "Imperialist competitive algorithm combined with chaos for global optimization", Communications in Nonlinear Science and Numerical Simulation, vol. 17, no. 3, pp. 1312–1319, 2012.

[46] T. Niknam, E. T. Fard, N. Pourjafarian, and A. Rousta, "An efficient hybrid algorithm based on modified imperialist competitive algorithm and K-means for data clustering", Engineering Applications of Artificial Intelligence, 24, 2, pp. 306–317, 2011.

[47] M. A. Ahmadi, M. Ebadi, A. Shokrollahi, and J. M. S. Majidi, "Evolving artificial neural network and imperialist competitive algorithm for prediction oil flow rate of the reservoir", Applied Soft Computing, vol. 13, no. 2, pp. 1085–1098, 2013.

[48] M. Hajihassani, D. J. Armaghani, A. Marto, and E. T. Mohamad, "Ground vibration prediction in quarry blasting through an artificial neural network optimized by imperialist competitive algorithm", Bulletin of Engineering Geology and the Environment, vol. 4, pp. 873–886, 2015.

[49] C. Lu, X. Li, L. Gao, W. Liao, and J. Yi, "An effective multi-objective discrete virus optimization algorithm for flexible job-shop scheduling problem with controllable processing times", Computers & Industrial Engineering, vol. 104, pp. 156–174, 2017.

 ISBN: 978-1-61208-882-2

Friction Invariant Object Reconstruction Using a Vibrissa-inspired Tactile Sensor Concept

Lukas Merker
Dept. of Mechanical Engineering
Technische Universität Ilmenau
Ilmenau, Germany
e-mail: lukas.merker@tu-ilmenau.de

Joachim Steigenberger†
Institute of Mathematics
Technische Universität Ilmenau
Ilmenau, Germany

Carsten Behn
Dept. of Mechanical Engineering
Schmalkalden University of Applied Sciences
Schmalkalden, Germany

†*In memory of Univ.-Prof. em. Dr. rer. nat. habil. Joachim Steigenberger (1933–2021).*
We miss his brilliant mind, his warm and humorous manner, his inspiring and constructive personality.

***Abstract*—Vibrissae-inspired sensor concepts hold potential of complementing conventional sensors due to their collision robustness and increased near-field scanning range. In general, they consist of a slender elongated probe, one-sided attached to some kind of measuring device. Making contact with an object, the probe gets bent and transduces (mechanical) signals to its support/the measuring device. Modeling this process analytically, most approaches in literature are based on the assumption of an ideal contact (no stiction or friction). Within the present paper, we extend an existing tactile sensor model for object contour scanning and reconstruction taking first steps to implement frictional effects (Coulomb's law of friction). The probe is modeled as a rod, which is one-sided clamped. Shifting the clamping position relatively to the object of interest, represented by a plane and strictly convex object contour, the rod sweeps along the contour quasi-statically. This process is analyzed in two steps: (1) simulating scanning sweeps in order to generate the support reactions (observables) at the base of the rod theoretically, assuming the object contour and the coefficient of friction to be known, (2) using the generated support reactions from the previous step in order to reconstruct a sequence of contact points, finally approximating the original object contour. Our simulations suggest, that friction affects the observables but (surprisingly) not the actual reconstruction error. Finally, we present a novel approach for reconstructing friction parameters based on the support reactions.**

Keywords–Vibrissa; tactile sensor; object scanning; object reconstruction; friction invariance

I. INTRODUCTION

Obstacle avoidance and object identification is a key topic in mobile robotics. Solving these tasks in our everyday life, we humans benefit from our sense of touch complementing visual information. In the same way, robots could benefit from advanced tactile sensors complementing their optical sensors. Beyond the human skin, another prominent natural example functioning as tactile sensors are vibrissae (whiskers) of rodents in animal kingdom. For instance, rats and mice are capable of localizing objects in space based on few vibrissal contacts [1]. As the vibrissal hair-shafts themselves consist of dead tissue, the animals solely rely on the signals sensed by receptors inside the support of each vibrissa, the Follicle-Sinus Complex (FSC) [2]. This biological principle has frequently been transferred to technical sensor concepts [3][4], which basically all share a common structure: a more or less flexible probe, mimicking the vibrissal hair-shaft, one-sided attached to some kind of measuring device, representing the FSC. Making contact with an object of interest, the probe gets bent to varying degrees. Measuring the support reactions (observables) at the base of the rod, conclusions can be drawn about the contact position in space. Ultimately, a whole sequence of contact points can be used to approximate the object's shape [5]. Instead of repeatedly tapping the object of interest with the probe as, e.g., in [6]–[10], one promising scanning strategy is to continuously sweep the probe along the object. The advantage of the sweeping procedure is its passive feasibility [11][12], e.g., using the robot movement as an actuation. Scanning objects via sweeping typically includes large deformations of the probe, whose analysis require nonlinear mechanical models. As such models rarely have solutions expressible in analytical terms, modeling approaches in literature frequently rely on numerics right from the beginning, e.g., [10][12]–[14]. However, these purely numerical approaches share the drawback of missing some important insights and analytical relations of the scanning and reconstruction process. In contrast to these largely numerical models, the authors of [15] presented a continuum model, consisting of a nonlinear Euler-Bernoulli bending rod, whose deformation equations were derived analytically. Using the measured support reactions at the base of the rod, the authors formulated an Initial-Value Problem (IVP). Only at the very end, numerical methods were used to integrate this IVP. In doing so, the contact position along the rod (and finally in space) was determined using a termination condition, that the bending moment at the contact point is zero. A similar model was used in [16] to analyze the problem of object scanning and reconstruction fully analytically in two (inverse) steps:

- step 1 – simulating scanning sweeps in order to generate the support reactions (observables) at the base of the rod theoretically, assuming the object contour to be known, and
- step 2 – using the generated support reactions from the previous step in order to reconstruct a sequence of contact points, finally approximating the original object contour.

Step 2 shows strong similarity to the approach in [15] but

 ISBN: 978-1-61208-882-2

differs by solving the problem fully analytically. Without any demonstration, the author of [16] claimed that the analytical procedure provides the same reconstruction quality as the numerical one used in [15]. However, as [16] is based on the assumption of ideal contacts (no stiction or friction), this statement should be viewed critically and verified using a more realistic contact model. This is where the paper at hand steps in, presenting a preliminary simulation-based study in the development of a more complex analytical vibrissa-inspired sensor model that allows to reconstruct friction and object shape information simultaneously.

The remainder of the paper at hand is structured as follows: In Section II, we present the vibrissa-inspired sensor model deriving the deformation equations of the probe in contact with an object. Following the procedure of [16], we separately analyze the above-mentioned steps 1 and 2 in Section II-A and Section II-B, respectively. Finally, Section III-C presents a novel approach of reconstructing friction parameters during object scanning. In Section III, simulation results are presented and analyzed with the overall goal of verifying some analytical key conditions of [16] in the presence of friction presenting a reconstruction benchmark between [15] and [16]. Finally, the results of the present paper are summed up and some future research subjects are identified in Section IV.

II. MODELING

A single highly flexible probe is modeled as a slender Euler-Bernoulli bending rod of length L consisting of a homogeneous and isotropic material. Its mechanical behavior is basically determined by a constant second moment of area I_z and a constant Young's modulus E. According to [16], we introduce the following units of measure to use only dimensionless system parameters:

$$[length] := L, \quad [force] := \frac{EI_z}{L^2}, \quad [moment] := \frac{EI_z}{L} \quad (1)$$

Considering the rod in a fixed (x, y)-coordinate system, we suppose the lower end of the rod ("foot, base") to be clamped at some point $(x_0, 0)$, see Figure 1. The process of object scanning is considered quasi-statically. It is realized by incrementally shifting the clamping position x_0 (input variable) relative to the object. Consequently, sweeping along the object contour, the rod gets bent and transmits mechanical signals to the clamping. The resulting support reactions f_{0x}, f_{0y} and m_{0z} at the clamping are considered as the observables. The object is assumed as a rigid body with a strictly convex contour function $g : x \mapsto g(x)$. Strict convexity is required for two reasons: Firstly, it prevents simultaneous multi-point contacts between the rod and the object [17]. Secondly, it allows to express the contour function in dependence on its slope angle $\tilde{\alpha} \in [-\frac{\pi}{2}, \frac{\pi}{2}]$ and, thus, we come up with the handy parametrization: $(x, g(x)) \mapsto (\xi(\tilde{\alpha}), \eta(\tilde{\alpha}))$ [16]. In contrast to [16], where the resulting contact force $\vec{f}$ is assumed to be perpendicular to the profile contour tangent, we do not assume such an ideal contact here. Instead, we consider the contact force $\vec{f} = \vec{f}_n + \vec{f}_t$ as composed of a normal component $\vec{f}_n$ and a tangential component $\vec{f}_t$ with the coefficient of friction $\mu = \frac{|\vec{f}_t|}{|\vec{f}_n|} = \tan(\zeta)$, where ζ denotes the friction angle. Then, inspired by Figure 2, the contact force writes:

$$\vec{f} = \vec{f}_n + \vec{f}_t = f[\sin(\alpha)\vec{e}_x - \cos(\alpha)\vec{e}_y] \quad (2)$$

with $\alpha = \tilde{\alpha} + \zeta$.

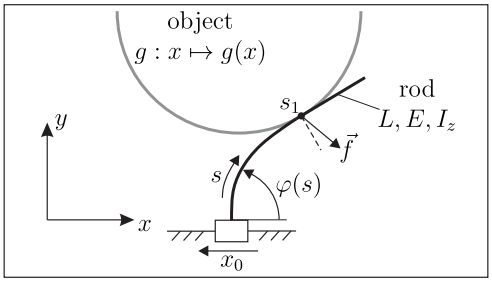

Figure 1. Mechanical model of a rod sweeping along a strictly convex object contour function.

Apart from this, the derivation of the modeling equations describing the elastic line of the rod is fully analogue with [16]. Therefore, it is roughly outlined here: Firstly, the elastic line is parameterized by means of its slope angle φ in dependence on its natural coordinate arc-length $s \in [0, 1]$. Using (2) in order to determine the bending moment $m(s)$, we take advantage of Euler's constitutive law to express the curvature $\kappa(s) = m(s)$ of the elastic line. Finally, introducing an additional differential equation for the curvature, we end up in a system of nonlinear Ordinary Differential Equations (ODEs) of first order:

$$\begin{array}{|ll|ll|} \hline \text{(a)} & x'(s) = \cos(\varphi(s)) & \text{(b)} & y'(s) = \sin(\varphi(s)) \\ \text{(c)} & \varphi'(s) = \kappa(s) & \text{(d)} & \kappa'(s) = f\cos(\varphi(s) - \alpha) \\ \hline \end{array} \quad (3)$$

Remark 1: It is important to understand, that, although (2) *and* (3) *have the same appearence as in [16], the angle $\alpha \geq \tilde{\alpha}$ is defined in a completely different way, no longer representing the slope angle $\tilde{\alpha}$ of the profile contour.*

A. Step 1: generating the support reactions theoretically

For generating the support reactions theoretically, we assume the object contour function as well as the coefficient of friction μ to be known. Within the present paper, we exemplarily consider an object contour function represented by the semi-circle

$$g : x \mapsto g(x) = y_c - \sqrt{r^2 - (x - x_c)^2} \quad (4)$$

with radius r and center coordinates x_c and y_c. During the scanning sweep, a distinction is made between tip and tangential contacts, see Figure 2:

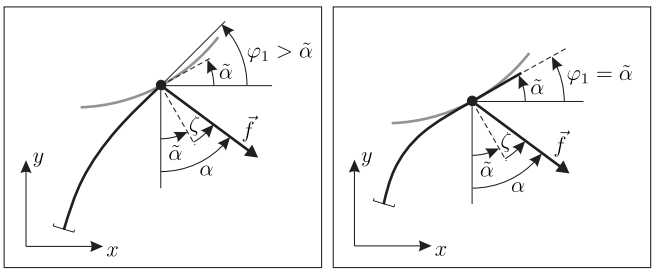

Figure 2. Comparison between (a) tip contact ($s = 1$), and (b) tangential contact ($s = s_1$) with angular relationships.

- For tip contacts, the position of contact along the rod $s_1 = 1$ is known, but the angle $\varphi_1 = \varphi(1) > \tilde{\alpha}$ is unknown
- In turn, for tangential contacts, the position of contact along the rod s_1 is unknown, but instead, we have the angular relationship $\varphi = \tilde{\alpha}$.

Finally, these observations result in the following Boundary-Conditions (BCs) at the clamping (5), for tip (6) and tangential contacts (7):

$$\boxed{\begin{array}{ll}\text{(a)}\ \ x(0) = x_0 & \text{(b)}\ \ y(0) = 0 \\ \text{(c)}\ \ \varphi(0) = \dfrac{\pi}{2} & \text{(d)}\ \ \kappa(0) = \kappa_0\end{array}} \tag{5}$$

$$\boxed{\begin{array}{ll}\text{(a)}\ \ x(1) = \xi(\tilde{\alpha}) & \text{(b)}\ \ y(1) = \eta(\tilde{\alpha}) \\ & \text{(c)}\ \ \kappa(1) = 0\end{array}} \tag{6}$$

$$\boxed{\begin{array}{ll}\text{(a)}\ \ x(s_1) = \xi(\tilde{\alpha}) & \text{(b)}\ \ y(s_1) = \eta(\tilde{\alpha}) \\ \text{(c)}\ \ \varphi(s_1) = \tilde{\alpha} & \text{(d)}\ \ \kappa(s_1) = 0\end{array}} \tag{7}$$

Remark 2: It is important to note, that (via α) ζ does affect the ODE system (3)*, but not the BCs* (6) *and* (7)*, see Figure 2. Instead, the BCs solely depend on the object slope $\tilde{\alpha}$.*

The process of generating the observables is realized by simulating scanning sweeps in Matlab 2019a. In doing so, the two-point boundary-value problems (3)&(5)&(6) and (3)&(5)&(7) are solved using shooting methods to determine the unknown parameters f_1, α, κ_0 and s_1. Finally, having the mentioned parameters at hand, the support reactions are determined in the following way:

$$f_{0x} = -f\sin(\alpha), \qquad f_{0y} = f\cos(\alpha), \qquad m_{0z} = -\kappa_0 \tag{8}$$

B. Step 2: Reconstructing contact points

For reconstructing a sequence of contact points, we assume the support reactions f_{0x}, f_{0y}, m_{0z} and the clamping position x_0 to be known in advance, either from step 1 or from measurements using a real experimental setup. Consequently, the unknown parameters f, α and κ_0 are determined evaluating (8) in the following way:

$$f = \sqrt{f_{0x}^2 + f_{0y}^2}, \quad \alpha = -\arctan\left(\frac{f_{0x}}{f_{0y}}\right), \quad \kappa_0 = -m_{0z} \tag{9}$$

Note, as α is not necessarily restricted to the domain $[-\frac{\pi}{2}, \frac{\pi}{2}]$ as in [16], the four-quadrant inverse tangent atan2 must be used in (9). Now, having all parameters of (5) at hand, we consider the IVP (3)&(5). As mentioned in Section I, the integration of this IVP can be realized numerically as in [15] or analytically as in [16]:

- In [15], a termination condition is used which cancels the numerical integration, if $\kappa(s_1) = m(s_1) = 0$ is fulfilled.
- In [16], the position s_1 is determined by solving the analytic expression using elliptic integrals

$$s_1 = -\frac{1}{\sqrt{2f}} \int_{\frac{\pi}{2}}^{\tilde{\alpha}} \frac{1}{\sqrt{\sin(t - \tilde{\alpha})}} dt \tag{10}$$

Both procedures (even the analytical derivation of (10)) are based on the fact, that the curvature at s_1 is zero. Therefore, it can be expected, that both procedures lead to the same results (except for minor numerical deviations). In fact, this hypothesis was made in [16], but without any proof or demonstration. Within the present paper, we take advantage of both reconstruction procedures comparing their results in a benchmark with a surprising result in Section III.

C. Reconstructing friction parameters

Assuming an ideal contact ($\zeta = 0 \Rightarrow \alpha = \tilde{\alpha}$) in [16], the problem of reconstructing the object slope $\tilde{\alpha}$ was trivial, as it was simply determined by the support reactions using (9). In the same way we are able to determine the orientation α of the contact force $\vec{f}$. However, here we face the problem that $\alpha = \tilde{\alpha} + \zeta$ includes two components – the object slope $\tilde{\alpha}$ and the friction angle ζ, see Figure 2. Without knowing the friction coefficient, it seems impossible, to separate α in its components. Adressing this problem, we use the curvature ODE (3)(d) together with (7)(d) to formulate the first integral

$$\frac{1}{2}\kappa(s)^2 = f(\varphi(s) - \alpha) - f(\varphi_1 - \alpha) \tag{11}$$

where $\varphi(s_1) = \varphi_1$. Now, substituting $s = 0$ and solving for φ_1 yields

$$\varphi_1 = \alpha - \arcsin\left(\frac{m_{0z}^2 - 2f_{0y}}{2f}\right) \tag{12}$$

$$= \tilde{\alpha} + \zeta - \beta \tag{13}$$

with $\beta := \arcsin\left(\frac{m_{0z}^2 - 2f_{0y}}{2f}\right)$. Analyzing (13), we again distinguish between tip and tangential contacts, see Figure 2:

- For tip contacts ($\varphi_1 > \tilde{\alpha}$), it can merely be concluded that $\beta < \zeta$.
- In contrast, for tangential contacts ($\varphi_1 = \tilde{\alpha}$), (13) can be rewritten in the following way:

$$\boxed{\zeta = \beta = \arcsin\left(\frac{m_{0z}^2 - 2f_{0y}}{2f}\right)} \tag{14}$$

Thus, in case of tangential contact, both the friction angle ζ and the coefficient of friction $\mu = \tan(\zeta)$ are determinable based on the support reactions. Then, the object slope is $\tilde{\alpha} = \alpha - \zeta$ and $\vec{f}$ decomposes into

$$\begin{aligned}\vec{f}_n &= f\cos(\zeta)\left[\sin(\tilde{\alpha})\,\vec{e}_x - \cos(\tilde{\alpha})\,\vec{e}_y\right] \\ \vec{f}_t &= f\sin(\zeta)\left[\cos(\tilde{\alpha})\,\vec{e}_x + \sin(\tilde{\alpha})\,\vec{e}_y\right]\end{aligned} \tag{15}$$

III. Results and Discussion

A. Generating the support reactions

Figure 3a shows an exemplary simulated scanning sweep of the rod along the circular object contour (4) with $r = 0.9$, $x_c = 0$ and $y_c = 1.3$ from right to left (negative x-direction). The sweep is represented by a sequence of elastic lines (equilibrium states), where tip contacts are colored in blue and tangential ones in red for $s \in (0, s_1)$ and black for $s \in (s_1, 1)$. The scanning sweep aborts at some point, where the rod would detatch from the object (snap-off). In general, the appearance

of Figure 3a does not change significantly with varying μ. Therefore, in Figure 3a we limit ourselves to only one example with $\mu = 0$. In Figs. 3b-3d, the support reactions f_{0x}, f_{0y} and m_{0z} for scanning sweeps with different friction coefficients $\mu = 0.0 : 0.2 : 0.6$ are plotted against the clamping position x_0. Additionally, the gray curve in each figure represents the support reactions resulting from a scanning sweep with a uniformly distributed random friction coefficient. In doing so, a random value $\mu \in [0, 0.4]$ was generated for each clamping position x_0 in order to model friction parameter uncertainty. In Figs. 3b-3d, the transitions between tip and tangential contacts (see Figure 3a) and vice versa are marked with a small circle. Note, that all Figs. 3a-3d are to be read from right to left, due to the scanning direction.

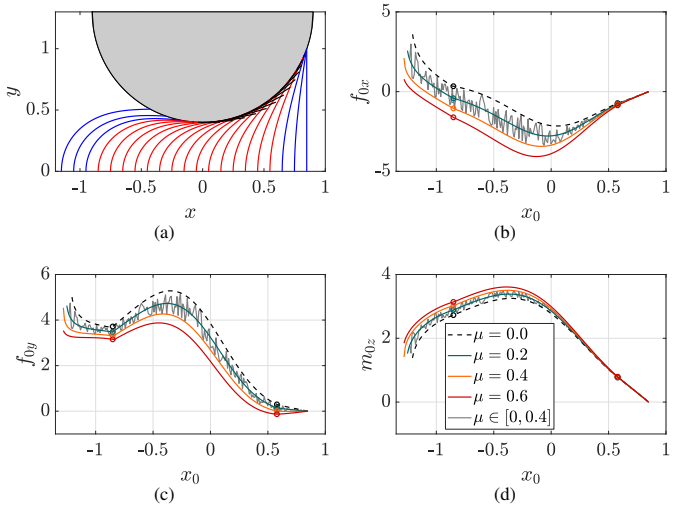

Figure 3. Object scanning: (a) sequence of equilibrium states during a scanning sweep (from right to left) along a circular object contour; (b)-(d) support reactions (observables) f_{0x}, f_{0y} and m_{0z} in dependence on different friction coefficients μ including a uniformly distributed random $\mu \in [0, 0.4]$ (gray line).

Based on Figure 3, we highlight the following observations:

- All support reactions are affected by the friction coefficient μ. For instance, the clamping moment m_{0z} consistently increases with increasing μ.
- Moreover, an increasing coefficient μ results in a longer overall contact phase, which is apparent comparing the values of x_0 at the end of each scanning sweep (left side of each curve).
- The friction coefficient μ has little impact on the transitions between tip and tangential contacts.

Summarizing, the simulation of step 1 allows for the generation of the support reactions in dependence on a pre-set (known) friction coefficient. Note that adjusting the friction coefficient to a defined value (as in the simulation) would be difficult to realize in an experiment. This highlights the actual importance of simulation step 1. Now, having the support reactions based on different friction coefficients μ at hand, this enables a comprehensive investigation of step 2. In the following, a comparison is made for the reconstruction procedures of [15] and [16] (see Section II-B) based on the support reactions in Figure 3b-3d.

B. Benchmark: Reconstructing contact points

As discussed in Section II, the basis for reconstructing contact points in space is the determination of the contact location s_1 along the rod. Here, we compare the approaches of [15] and [16] (see Section II), starting with an evaluation of s_1 in Figs. 4a and 4b. In Figure 4a, s_1 is determined based on [15] using an event-function, which cancels the numerical integration, when the curvature of the rod is zero for the first time. It is obvious, that all curves $s_1(x_0)$ are quite close to each other, and therefore, the friction coefficient seems to have little impact on s_1 during object scanning: the enlarged section shows, that an increasing friction coefficient generally results in a slightly larger s_1. By analogy with Figs. 3b-3c, the gray curve of s_1, resulting from a random friction coefficient, oscillates between the orange and the black one.

In Figure 4b, s_1 is determined based on [16]. Evaluating the support reactions using (10), two main observations were made:

- Considering the case $\mu = 0$ (black curve), the fact that (10) returns values larger than 1 in the presense of tip contacts, as stated in [16], is correct for a large extend of the scanning sweep. Note that all values exceeding 1 are replaced by the value $s_1 = 1$ (tip of the rod) in Figure 4b. However, one exception, where (10) returns values lower than 1, even though the rod is in tip contact, is observed at the very end of the scanning sweep (see black curve in Figure 4b). Hence, using (10) as a decision criterion between tip and tangential contact is not adequate.
- For $\mu > 0$, (10) does no longer provide any information about the contact phase at all, see Figure 4b. Moreover, from a certain threshold (in our case $\mu \geq 0.4$), the arguments exceed the definition range of (10) for some clamping positions x_0. For instance, the orange and red curves are not defined at the beginning of the scanning sweep, see Figure 4b.

Comparing Figs. 4a and 4b it is obvious, that, in the absence of friction (black curves), the reconstructed contact positions s_1 along the rod are almost identical. However, there seems to be a huge difference in the reconstruction procedures of [15] and [16], which only becomes noticeable in the presense of friction. The origin of this difference lies in the derivation of the analytical condition (10). There, the condition $\varphi = \alpha$ is used assuming an ideal, tangential contact. And exactly that is the problem: The condition $\varphi = \tilde{\alpha} = \alpha$ is indeed valid for the case $\mu = 0$ (ideal contact), but results in increasing errors with increasing μ: $\varphi = \tilde{\alpha} = \alpha - \mu$. In contrast, the reconstruction procedure of [15] dispenses with the mentioned geometric condition. Therefore, it is not affected by increasing friction coefficients. Note that the slight deviations in Figure 4a are not caused by any reconstruction error, but instead they reflect the actual change of s_1 caused the change of μ. Beyond the contact position s_1, the actual reconstructed sequences of contact points are shown in Figs. 4c and 4d. Again, Figure 4c is based on the reconstruction procedure presented in [15] while Figure 4d is based on [16]. In both cases, the reconstructed sequences of contact points are superimposed by the original object contour (gray line). The corresponding reconstruction

 ISBN: 978-1-61208-882-2

errors, which are defined as the smallest (perpendicular) distance to the original object contour, are indicated in Figs. 4e and 4f.

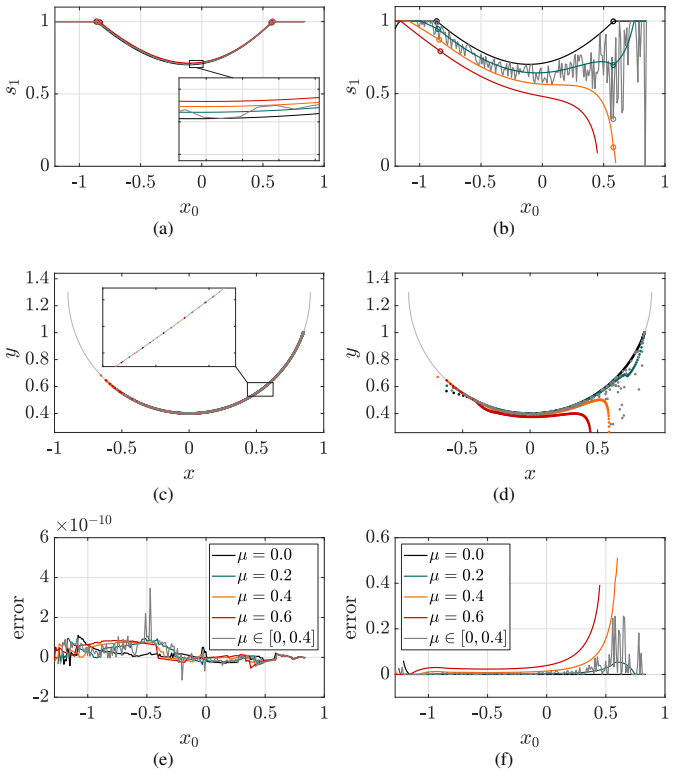

Figure 4. Object reconstruction using the support reactions (observables) from Figs. 3b–3d: (a) reconstruction of s_1 based on [15]; (b) reconstruction of s_1 using (10) based on [16]; (c) reconstructed contact points based on [15]; (d) reconstructed contact points based on [16]; (e) reconstruction error corresponding to (c); (f) reconstruction error corresponding to (d).

The following two observations must be highlighted:

- Most noteworthy, independently on μ, the reconstructed contact points using the procedure of [15] lie exactly on the original object contour, see enlarged section of Figure 4c. This fact reflects in Figure 4e, where the reconstruction errors are within numerical boundaries for all friction coefficients μ, including the randomly generated one. This is probably the most important observation of the presented paper. With increasing μ, the reconstructed contact points shift along the object contour while the reconstruction error remains unchanged. This is what we refer to as friction invariant reconstruction.
- In contrast, the reconstructed points in Figure 4d are characterized by larger errors. For $\mu = 0$, the reconstruction error in Figure 4d (black line) is in the same order of magnitude as those in Figure 4c, except for the end of the scanning sweep. In general, the reconstruction error increases with increasing friction coefficient μ. The reconstruction based on the randomly generated μ is characterized by many outliers. Summarizing, being not friction invariant, the reconstruction error of s_1, see Figure 4b, reappears in the reconstruction of the contact points in Figure 4d.

C. Reconstructing friction parameters

Besides the reconstructed contact points, it can be advantageous in many ways to know the normal direction of an object – on the one hand, normal directions might assist in fitting a curve through he reconstructed contact points, improving the reconstruction quality. On the other hand, they might be evaluated for object manipulation, e.g., grasping. In the following, we use the observables from Figs. 3b–3d in order to verify the analytical condition (14). The angle $\beta = \alpha - \varphi_1$, see (12), is plotted against the clamping position x_0 in Figure 5a, where the phase transitions are indicated by circular markers. According to Section II, the condition $\beta = \zeta$ holds in case of tangential contact, i.e., in the interval enclosed by the markers in Fig. 5a, where the angle β is constant. In order to evaluate the data, the coefficient of friction $\mu = \tan(\zeta)$ is determined based on the reconstructed friction angle ζ. For the tangential contact phase, $\mu(x_0)$ is shown in Fig. 5b.

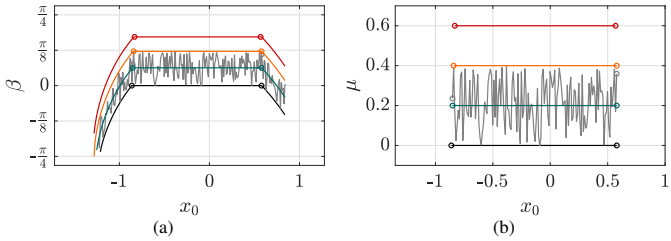

Figure 5. Reconstructing friction parameters: (a) angle $\beta = \alpha - \varphi_1$ plotted against the clamping position x_0 (phase transitions marked with circles); (b) friction coefficient $\mu = \tan(\zeta)$ for tangential contacts.

It can be seen that the reconstructed friction coefficients are consistent with the values originally assumed for μ. Even the noisy function for $\mu \in [0, 0.4]$ matches the pre-set one in simulation step 1. The observation confirms the hypothesis that, for tangential contacts, the contact force $\vec{f}$ can be decomposed into its normal and tangential components by means of (15), see Section II.

IV. Conclusion

The paper at hand presented a mechanical model of a vibrissa-inspired tactile sensor for object contour scanning and reconstruction in the presence of Coulomb friction. In doing so, two different processes were analyzed separately: Firstly, scanning sweeps along a known object contour assuming a pre-set friction coefficient were simulated in order to generate the support reactions (observables) theoretically. Afterwards, these support reactions were used in order to reconstruct a sequence of contact points. In doing so, the reconstruction approaches of [15] and [16] were compared in a benchmark test. Finally, a novel approach of determining friction parameters based on the support reactions was presented, resulting in a handy analytical expression. Summarizing the results of the paper at hand, we highlight the following observations: During object scanning, dry friction affects the support reactions. It turned out that the reconstruction approach presented in [15] is friction invariant, i.e., friction does not affect the reconstruction error. In contrast, the approach presented in [16] is applicable just as well as the one presented in [15] in the absence of friction, but results in large reconstruction errors in the presence of friction.

 ISBN: 978-1-61208-882-2

However, even though the approach in [16] is not friction invariant, it laid an important foundation for all investigations of the present paper. By identifying the essential differences between the mentioned two approaches, it is expected that the analytical conditions from [16] can be adapted in future work to be friction invariant as well. Finally, the novel approach of detecting friction parameters based on the support reactions was verified by simulations.

The mentioned findings suggest that the presented measuring principle is highly suitable to complement optical sensors in robot exploration and path planning tasks. For instance, a robot equipped with one (or multiple) vibrissa-like sensors might move through the environment, passively dragging the highly flexible probe. After accidentally making a soft contact between some point of the probe and an object/obstacle in the environment, the probe would get bent, causing the support reactions to exceed a certain activation threshold. Subsequently, the measured signals might be used to draw further conclusions about the object's shape and ultimately to assist in path-planning algorithms, tracking a prescribed value of the support reactions to achieve an optimal scanning movement of the robot, relative to the object.

The research conducted in this paper is particularly limited by the use of a quasi-static model. Against this background, only static friction but no dynamical effects, e.g., stick-slip effects can be discussed. However, the fact that even a randomly generated (noisy) friction coefficient did not invalidate the investigations suggests, that the results in the paper at hand also apply to practical scenarios. Finally, it remains to verify the theoretical results on practical examples by using an experimental setup which has already been attacked in the first steps.

REFERENCES

[1] T. J. Prescott, B. Mitchinson, and R. A. Grant, "Vibrissal behavior and function," Scholarpedia, vol. 6, pp. 6642, 2011.

[2] S. Ebara, T. Furuta, and K. Kumamoto, "Vibrissal mechanoreceptors," Scholarpedia, vol. 12, pp. 32372, 2017.

[3] F. A. Lucianna, A. L. Albarracín, S. M. Vrech, F. D. Farfán, and C. J. Felice, "The mathematical whisker: A review of numerical models of the rat's vibrissa biomechanics," J. Biomech., vol. 49, pp. 2007–2014, 2016.

[4] C. Behn, "Mathematical Modeling and Control of Biologically Inspired Uncertain Motion Systems with Adaptive Features," Habilitation, Technische Universität Ilmenau, Germany, 2013.

[5] C. L. Schroeder and M. J. Z. Hartmann, "Sensory prediction on a whiskered robot: a tactile analogy to optical flow," Frontiers in Neurorobotics, vol. 6, 2012. DOI: 10.3389/fnbot.2012.00009.

[6] T. Tsujimura and T. Yabuta, "Object detection by tactile sensing method employing force/torque information," IEEE Trans. Robot. Autom., vol. 5, pp. 444–450, 1989.

[7] M. Kaneko, N. Kanayama, and T. Tsuji, "Active antenna for contact sensing," IEEE Trans. Robot. Autom., vol. 14, pp. 278–291, 1998.

[8] J. H. Solomon and M. J. Z. Hartmann, "Artificial whiskers suitable for array implementation: Accounting for lateral slip and surface friction," IEEE Trans. Robot., vol. 24, pp. 1157–1167, 2008.

[9] J. A. Birdwell et al., "Biomechanical models for radial distance determination by the rat vibrissal system," J. Neurophysiol., vol. 98, pp. 2439–2455, 2007.

[10] D. Kim and R. Möller, "Biomimetic whiskers for shape recognition," Robot. Auton. Syst., vol. 55, pp. 229–243, 2007.

[11] L. Merker, J. Steigenberger, R. Marangoni, and C. Behn, "A vibrissa-inspired highly flexible tactile sensor: scanning 3D object surfaces providing tactile images," Sensors, vol. 21, 2021. DOI: 10.3390/s21051572.

[12] J. H. Solomon and M. J. Z. Hartmann, "Extracting object contours with the sweep of a robotic whisker using torque information," Int. J. Robot Res., vol. 29, pp. 1233–1245, 2010.

[13] B. W. Quist, R. A. Faruqi, and M. J. Z. Hartmann, "Variation in Young's modulus along the length of a rat vibrissa," Journal of Biomechanics, vol. 44, pp. 2775–2781, 2011.

[14] L. A. Huet, J. W. Rudnicki, and M. J. Z. Hartmann, "Tactile sensing with whiskers of various shapes: determining the three-dimensional location of object contact based on mechanical signals at the whisker base," Soft Robotics, vol. 4, pp. 88–102, 2017.

[15] G. R. Scholz and C. D. Rahn, "Profile sensing with an actuated whisker," IEEE Trans. Rob. Autom., vol. 20, pp. 124–127, 2004.

[16] C. Will, "Continuum Models for Biologically Inspired Tactile Sensors: Theory, Numerics and Experiments," Ph.D. Thesis, Technische Universität Ilmenau, Germany, 2018.

[17] L. Merker, S. J. Fischer Calderon, M. Scharff, J. H. A. Miranda, and C. Behn, "Effects of Multi-Point Contacts during Object Contour Scanning Using a Biologically-Inspired Tactile Sensor," Sensors, vol. 20, 2020. DOI: 10.3390/s20072077.

 ISBN: 978-1-61208-882-2

Digital Twin for Drone Control through a Brain-Machine Interface

Diana Ramos
Faculdade de Engenharia da Universidade do Porto
Capgemini Engineering
Porto, Portugal
email: dianacristina.teixeiraramos@altran.com

Gil Gonçalves
Faculdade de Engenharia da Universidade do Porto
Porto, Portugal
email: gil@fe.up.pt

Ricardo Faria
R&D Tech Leader
Capgemini Engineering
Lisboa, Portugal
email: ricardoandre.pintofaria@altran.com

***Abstract*—Drones enable humans to perform certain high-risk and attention operations and safety-critical tasks remotely, which are boosted by the use of Brain-Computer Interfaces. However, these technologies are correlated with the cognitive state of the operator, who is prone to stress and diversions, which brings instability to drone control. In this paper, we propose a decision making system aiming to decide, upon the operator's emotional state, whether the command should or should not be sent to the drone. By building a predictive operator's digital twin for cognitive emotional detection and by benefiting from a visual facial expression classifier, this system computes the coordinates and sends them to the drone through a Robot Operating System 2 client. Results show that both the digital twin and the facial expression classifier are capable of detecting emotions in a real-time setting and the system provides a reliable and secure way of commanding drones through the mind.**

***Keywords*—drone; Brain-Computer Interface; digital twin; Robot Operating System 2.**

I. Introduction

The drone sector has been growing with higher demand through the years. The common belief is that drones are singularly used for military affairs; however, they are functional and versatile systems. One major use is providing monitoring services, i.e., target searching, surveillance for security purposes and others. Even thought they have attracted companies due to their visionary application, the most significant change is how civilians have been incorporating them for entertainment purposes or as assistive devices to help with their routines [1]. Still, the most impactful usage of drones is their applicability to complete high-risk and safety-critical operations with success, often in locations unreachable and/or dangerous to humans.

A. Problem Overview

Controlling one drone is already a complex task. Operators are responsible for, not just to perform standard operations (takeoff and landing) with success, but also to safely execute them. When adding unsafe and critical operations to the task log, the control complexity increases significantly. The operator needs abilities at their peak, full attention/focus when performing these operations, to provide a reliable and stable control.

Hand control allows operators to remotely send commands to drones; however, as these are critical systems, operators need to be cautious with the commands they deliver. The Brain-Computer Interface (or BCI), or Brain-Machine Interface, is an alternative control mechanism. As humans are prone to fatigue, increasing mental workload and emotions, the control will become uncertain and insecure, that is, the operator can potentially mislead the drone with the wrong commands. In addition, BCIs require operators to have previous experiences to learn to formulate commands, which implies multiple training sessions. Even so, these command classifications are error-prone and contribute to unreliable control.

B. Proposed Solution

The hypothesis of this work is that by adopting a digital twin [4] to virtually represent the operator and by using machine learning techniques, it is possible to process, filter and predict whether the human operator has high mental workload and/or impactful emotions and decide whether the commands produced by the operator should or should not be sent to the drones. With the goal of validating the formulated commands, the digital twin is complemented with a visual emotion recognizer that will classify the operator's visual facial expression into a set of emotional states. Additionally, a Robot Operating System 2 (or ROS2) client node can be used in order to send the commands to the drone.

II. State-of-Art

A BCI is defined as "a device that connects the brain to a computer and decodes in real time a specific, predefined brain activity" [2]. This technology can use direct or indirect methods to do so, namely by evaluating the nerve cells activity or by assessing the levels of blood oxygen for these cells [2]. This technology has proved its relevance in many areas, for instance, there was a study aiming to deliver accurate real-time and precise command classification for drone reliable control. An Electroencephalography (or EEG) headset was used to record the brain activity, followed by a motor imagery acquisition. This mechanism involved four tasks, based on the subject visualizing physical movements instead of performing

 ISBN: 978-1-61208-882-2

them. Then, a classification methodology was developed by combining the Common Spatial Paradigm (or CSP) and the Linear Discriminant Analysis algorithms (LDA) [3]. Using this method, the authors were able to improve classification precision in real time. The solution was validated using a fixed-wing drone use case [3].

Another crucial component of this work is the digital twin. Nowadays, a virtual twin is described as a virtual representation that carries information to realistically behave and change as a physical hardware [4]. This technology is constantly evolving to serve each project needs. One variant that derives from it is the digital twin environment [4] with predicting capabilities. The main goal is to train the digital twin to gain predictive capabilities in order to anticipate the hardware's response or behaviour in situational events during run time. One example is a research work that proposes a framework to improve the estimates of certain measurements of physical systems, more specifically a drone, by implementing a virtual layer, i.e., a digital twin, that would represent the real device and predict its performance [5]. This approach implies that each piece of the drone has its own prediction models that should learn and be updated through time to, ultimately, accurately anticipate some metrics that are valuable to the end-user.

III. Implementation

As a starting point, we chose the *Emotiv Epoc+* headset for data acquisition due to its portability and reliability as a commercial BCI. In addition, this headset is connected with an application, *EmotivBCI*, that aids users as it provides a platform for direct command and facial expression's training and monitoring of multiple data streams. For accurate detection of command patterns, the application is supported by a machine learning prediction model to build a profile and refine it each time the user has a training session. For the purpose of this work, it was necessary that the operator was subjected to multiple sessions of command training to ensure its accuracy. After this stage, the operator was able to formulate *right* and *left* commands, and establish a *neutral* one (stationary state).

As illustrated by Figure 1, this system is composed by 4 components: (1) the digital twin, (2) the visual classifier/component, (3) the decision making component and (4) the ROS2 component.

The digital twin is a virtual representation of the operator and its goal is to classify, in real-time, the operator's emotional state. It relies on data collected by the BCI to build a cognitive profile, adapted to the operator. It is the core component of the proposed solution and provides decisive information to ascertain the destination of the command. The remaining components that follow are designated to support the digital twin and add complementary information for the decision. Three of the data streams, recorded by the *Emotiv Epoc+* headset, are collected: the band power, i.e., power of the EEG data according to the sensor and frequency band; the motion, based on the built-in gyroscope of the headset and the facial expressions, recorded from facial muscle motions. The system communicates with the cortex API to send requests for these data subscriptions and receive JSON responses with the resulting data streams as well as the classified commands, for the time period of the subscription. Since each request and response are unique to each stream, the newly collected data is integrated according to the nearest point in time of each observation, resulting into a single dataset. Columns with unique values are eliminated from this dataset, as well as features that do not add any value for the resolution of the problem. Motion related-features (categorical data) are transformed into binary columns, representing each type, through an one hot encoder. In addition, 2 features were added to the dataset: *arousal* and *valence* values, that are computed according to certain values of band power (according to [6]).

For the classification of the operator's emotional states, a set of classes were selected to represent positive and negative states. The positive classes are *calm* and *focused*, representing a stable cognitive state to send commands to the drone, as opposed to the negative classes (i.e., *stressed* and *distracted*) that detail an unstable cognitive state and, therefore, unacceptable state to send commands. In this work, the same operator simulated all the four emotions, at multiple days, in sessions of 8 seconds, reproducing a balanced dataset of about 19500 observations per emotion. In this work, we split 70% of the data for training the algorithms and 30% for testing and evaluated 8 classifiers in 4 performance metrics. As presented in Table I, Random Forest outperforms the remaining algorithms and is chosen for the training and modeling of the digital twin.

TABLE I
Evaluation of Algorithms

Algorithms	Performance Metrics			
	Accuracy	***Precision***	***Recall***	***f1-Score***
Decision Tree	0.995	0.995	0.995	0.995
k-NN	0.997	0.997	0.997	0.997
LDA	0.911	0.916	0.911	0.912
Naive Bayes	0.614	0.645	0.614	0.617
Random Forest	0.999	0.999	0.999	0.999
Support Vector Machine (or SVM) (linear kernel)	0.994	0.994	0.994	0.994
SVM (rbf kernel)	0.888	0.923	0.888	0.894
Neural Networks	0.948	0.949	0.948	0.948

Regarding the visual component of the system, a camera captures the real-time image of the operator and uses a Convolutional Neural Network based-prediction model [7] from an open-source project [7], modeled and trained with the FER-2013 emotion dataset, to classify the visual expressions of the operator as a set of emotions. This component can output: *positive* emotions as *happy* and *neutral* and *negative* emotions as *angry, disgust, fear, sad and surprise*.

While the *EmotivBCI* application classifies the reproduced commands from the operator, the digital twin receives information from the headset and classifies the cognitive state of the operator. The visual component receives the image from the camera and classifies the facial expressions. This process results in three input variables for the decision component. This decision module will decide whether the operator is stable

 ISBN: 978-1-61208-882-2

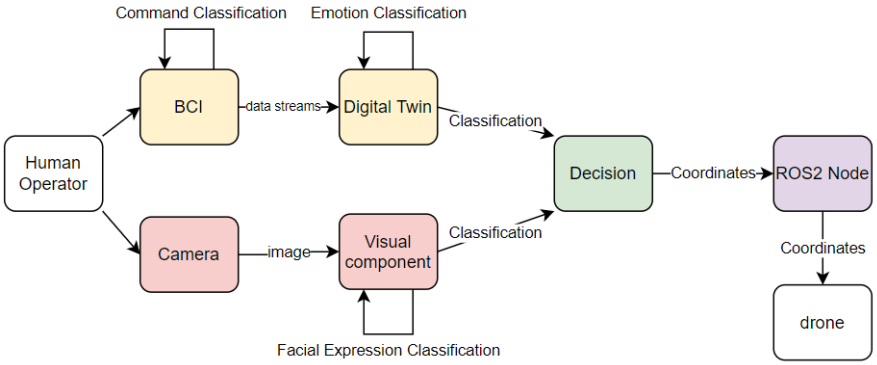

Figure 1. System architecture.

by mentally and visually evaluating his state. Only *positive* emotions detected on both components will allow the operator to send the command. Considering the confidence percentage of the command and the digital twin upon the classification, the decision module, in case of an overall positive emotion detection, will compute the drone coordinates accordingly and send them through a ROS2 node.

For the connection between the system and the drone, a ROS2 client-server architecture is created between what is called the *base station*, meaning the server machine that manages the drone, and the client node that sends requests through a service scheme, specific to a certain operation (takeoff, relative motions and landing). The client node is implemented as a gateway of the decision module, sending a request with the coordinates; the server receives the coordinates and forwards them to the drone in real-time.

IV. Experiments

It is expected that, after the operator training session and digital twin training, the system is capable of detecting multiple emotional states of the operator in real-time and handle the drone accordingly. So, to validate this approach, it was assembled a physical environment for secure and controlled drone flight demonstrations, composed by a four square meter indoor zone, called the *arena*, and a pair of *anchors* on each vertex, forming a positioning system that locates the drone by referencing its absolute coordinates. This *arena* was used for operating a *crazyflie* quadcopter in real-time.

To evaluate the different impacts of the solution, functionalities were split in a multi-level manner that go from the lowest experiment to the highest level (solution as a whole) to emphasize its value on securing a stable control environment for the drone. These experiments are: (1) the *baseline test*, defining the current state of drone control without the support of emotion recognition, (2) the *level 1 test*, representing the implementation of the core of the solution which is the cognitive emotion recognition system, (3) the *level 2 test*, the same as the previous test but with the addition of the computation of distances according to the confidence of both the mental command and classified cognitive state of the subject and (4) the *full test*, having all the above functionalities and the support of the visual emotion recognition.

With the exception of the *baseline test*, which gives no importance to the mental state of the subject, each test covers the four mental states (*focused, calm, distracted and stressed*) individually, each one with sessions of 8 seconds. The subject had to be put under the same conditions in which he used to simulate the four emotions on the training phase.

V. Results and Discussion

Given the environment set-up described in Section IV, the number of observations per emotion and per experiment, for the same subject that trained the commands in the *EmotivBCI* application, are described in Table II:

TABLE II
Number of Observations per Emotion

Emotions	**Group of Test**		
	Level 1 Test	***Level 2 Test***	***Full Test***
Calm	142	120	85
Focused	94	101	90
Distracted	124	135	104
Stressed	134	120	112

From the number of observations, it was computed the success rate, or accuracy, for each emotion and per experiment (Figure 2). This metric is calculated by dividing the number of correctly classified observations by the total amount of observations. For the *calm* state, the highest accuracy of the digital twin was 87,5%, for the *focused* state a 98,8%, for the *distracted* state a 93,5% and for the *stressed* state a 100%, as described by the round values on Figure 2.

Even with a high average of success rate for detecting the subject's mental states, the most accurately classified emotion was the *stressed* state. The difference between them can be due to the distinct way the model is trained in this segment, which involves more physical movement to denote agitation, rather than a low on motion condition on the remaining ones.

Lower success rate depicted on *level 1* for the *focused* state can be explained by the different background noise and movement between the training and test phase. This caused the subject to deviate his attention, explaining the occurrences of *distracted* classifications during this period. In the next test levels, this value is no lower that 80%, which is explained by the calmer environment. As opposed to this situation, the lower

 ISBN: 978-1-61208-882-2

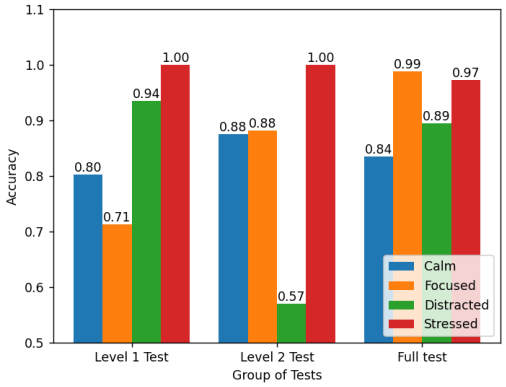

Figure 2. Success rate bar chart.

success rate on *level 2* for the *distracted* state classification can be explained by the lower amount interference or other diversions derived from background movement which led to short occurrences of focus by the subject.

Regarding the classification of the negative emotional spectrum (*distracted* and *stressed* states), Tables III and IV give some insight about the number of sent commands under an incorrect classification.

TABLE III
DISTRACTED EMOTION RECOGNITION

Positive Detections	**Group of Test**		
	Level 1 Test	***Level 2 Test***	***Full Test***
Total number	6	11	10
Nº of neutral commands	4	7	6
Nº of sent commands	2	4	1
BCI positive, visual negative	N/A	N/A	3

As registered in Table III, at *level 1* were detected 6 positive states, 2 of them sent; at *level 2* were detected 11 positive emotions, 4 were sent and at the *full test*, 10 positive emotions were detected, 1 command was sent to the drone and 3 were prevented due to the detection of a negative emotion by the visual component.

TABLE IV
STRESSED EMOTION RECOGNITION

Positive Detections	**Group of Test**		
	Level 1 Test	***Level 2 Test***	***Full Test***
Total number	0	0	2
Nº of neutral commands	0	0	1
Nº of sent commands	0	0	0
BCI positive, visual negative	N/A	N/A	1

As registered in Table IV, at the *full test* were detected 2 positive emotions and none were sent to the drone. One of them was a neutral command and the other was associated with a negative visual emotion, detected by the visual emotion component.

Since the training of mental commands is a task that requires time to practice and refine, it is challenging to reproduce a command at a live setting and in an equivalent environment the subject trained. Even with a digital twin inaccurate classification, most commands detected by the BCI are neutral ones, which have no impact on the trajectory of the drone. However, the command classifier can incorrectly output a *right* or *left* commands and these can potentially be sent to the drones. With the extra layer of the visual component, these unique situations are assessed by it and some of those errors are prevented. At a mission environment, where the operator needs to follow a sequence of commands, if there is a cancellation of a certain command, the operator will observe it and has enough time to reproduce the needed operation.

Considering that this is a 4-class classification problem, there is a probability of 25% that a baseline classifier correctly categorizes the subject emotion state and, in the *baseline test* characterized by the lack of machine learning, all commands are sent to the drones, regardless of the operator's emotional state, which could only be beneficial if the subject has perfect cognitive condition at all times.

VI. CONCLUSION

In this work, we analysed EEG data captured by the BCI *Emotive Epoc+* of a drone operator and, using machine learning techniques, we were able to build a digital twin of the operator capable of predicting its emotional state and decide whether the commands should be sent to the *crazyflie* quadcopter. The classification of the emotional state not only is supported by EEG data but also by a visual component that analyses the facial expressions. In addition, the communication between the system and the drone is done through a ROS2 client node. Multiple machine learning algorithms were validated and Random Forest was the best fitted and therefore used for training the digital twin. Results showed that the digital twin can accurately discriminate the operator's emotional states at a live setting and the combination of classification models improved the security and reliability of the system to decide upon the broadcasting of the reproduced commands. In the future, the goal is to adapt the current system to a swarm of drones, improving the training of the mental commands, the digital twin's accuracy and the efficiency of the digital twin's training and validate this approach with a larger number of subjects with different demographics.

REFERENCES

[1] A. Kreilinger et al., "Switching between manual control and brain-computer interface using long term and short term quality measures", *Frontiers in Neuroscience*, vol. 5, p. 147, 01 2011.

[2] A. Kubler, "The history of bci: From a vision for the future to real support for person hood in people with locked-in syndrome", *Neuroethics*, vol. 13, no. 2, p. 163–180, 07 2020.

[3] R. M. Vishwanath, S. K. Saksena, and S. Omkar, "A real-time control approach for unmanned aerial vehicles using brain-computer interface", *CoRR*, vol. abs/1809.00346, 2018.

[4] M. Grieves, "Origins of the digital twin concept", Available online: https://www.researchgate.net/publication/307509727_Origins_of_the_Digital_Twin_Concept, 08 2016.

[5] H. Y. Jeon, C. Justin, and D. Mavris, "Improving prediction capability of quadcopter through digital twin", in *AIAA Scitech 2019 Forum*, p. 1365, 01 2019.

[6] R. Ramirez and Z. Vamvakousis, "Detecting emotion from eeg signals using the emotive epoc device", in *Brain Informatics*, vol. 7670, pp. 175–184, 12 2012.

 ISBN: 978-1-61208-882-2

[7] U. Gogate, A. Parate, S. Sah and S. Narayanan, ”Real Time Emotion Recognition and Gender Classification”, *2020 International Conference on Smart Innovations in Design, Environment, Management, Planning and Computing (ICSIDEMPC)*, pp. 138-143, doi: 10.1109/ICSIDEMPC49020.2020.9299633, 2020.

 ISBN: 978-1-61208-882-2

Controlling Individual and Collective Information for Generating Interpretable Models of Multi-Layered Neural Networks

Ryotaro Kamimura
Kumamoto Drone Technology and Development Foundation
Techno Research Park, Techno Lab 203
1155-12 Tabaru Shimomashiki-Gun Kumamoto 861-2202
and IT Education Center, Tokai Univerisity
4-1-1 Kitakaname, Hiratsuka, Kanagawa 259-1292, Japan
email: ryotarokami@gmail.com

***Abstract*—The present paper aims to control selective information to understand the main mechanism of information processing in multi-layered neural networks. We propose two types of selective information, namely, individual and collective selective information, or simply, individual and collective information. The individual information represents to what degree a neuron is connected specifically to another one, and it should be increased as much as possible. Then, we try to use this abundant information as impartially as possible, reducing the specificity of collective neurons and reducing collective information. By controlling the ratio of individual and collective information, we can realize a number of different types of states to be interpreted, leading to the interpretation of the inference mechanism. The method was applied to the bankruptcy data set. In the experiments, we successfully increased individual information and decreased collective information. By examining partially compressed weights, we could see how neural networks, by controlling the selective information, can process information content in multi-layered neural networks. This examination of information flow can lead us to understand the main inference mechanism of neural networks.**

***Keywords*—individual, collective, information, selectivity, partial compression, interpretation, generalization**

I. INTRODUCTION

The black-box property of neural networks, as well as machine learning have caused much confusion in their applications, as well as model evaluation [1]–[6]. The black-box presupposition for human intelligence has been one of the major concerns in the possible introduction of machine learning into our society [7]–[10]. From a certain viewpoint, the black box is necessary, because the studies on human intelligence have been so far very limited and premature. However, we think that this confusion seems to be due to the different types of objectives in model creation and application. When we try to model human intelligence, our nervous systems, and cognitive systems in terms of neural networks, it is absolutely necessary to understand and interpret the main inference mechanism inside. The objective of these types of studies is to reduce the black-box properties as much as possible. This approach was very active in the early stage of development of neural networks, as has been well known in terms of connectionism [11]–[14]. The neural networks were considered models to create new information or knowledge by which we could deepen the understanding our nervous and cognitive systems.

On the contrary, if we try to apply models to practical problems, the objective is not necessary to understand human intelligence or cognitive processes, but to apply well and appropriately the models to practical problems naturally. From this point of view, the recent development of neural networks, as well as machine learning seem to be focused on the application to the practical problems. For example, the Convolutional Neural Networks (CNN) have been greatly developed recently, but they have been based on the simplified models of visual nervous systems [15]–[18], developed in the early seventies. However, the simplicity of the models, inherited only partially from the properties of our nervous systems [18], has made it possible to improve their prediction performance in an unexpected way. In other words, they have tried to extend some parts of our visual nervous systems to many different types of practical applications, keeping the main mechanism of optical systems only partially known.

As mentioned above, those methods have aimed not to understand the main nervous systems but to improve recognition and prediction performance, and the black-box property has been not so serious as had been expected. Naturally, there have been many attempts to interpret the inference mechanism in the field of convolutional neural networks due to the urgent need to respond to the right of explanation [9]. However, the majority of methods have been focused on the individual interpretation of neural networks for specific input patterns. Many attempts have been made to examine what kind of features among given input patterns are extracted in components of neural networks [19]–[26]. Since we have not known the main inference mechanism inside neural networks, all we can do is to uncover partially and step by step the characteristics related to specific input patterns. It should be repeated that this prevailing approach is very natural, because the convolutional neural network itself does not aim to clarify the human visual nervous framework itself, but it tries to enhance the simple and easily accessible parts of nervous systems as much as possible to strengthen the model performance itself.

Thus, we can say that one of the main problems of interpretation is that the objects to be interpreted have been ambiguous. If our objective is to apply the model to practical applications, the present models of interpretation, prevailing in the convolutional neural networks, may be sufficient. But if we try to understand the main mechanism of human intelligence and cognitive processes in terms of neural networks, we should think of a different type of interpretation.

Actually, there were serious attempts in neural networks to understand the inference mechanism of living systems, namely, the information-theoretic methods. One of the most

 ISBN: 978-1-61208-882-2

important approaches is the maximum information preservation by Linsker [27]–[30]. He tried to understand the visual nervous mechanism by supposing that neural networks try to increase mutual information as much as possible. Then, from this maximum information principle, he tried to explain the actual phenomena observed in living systems. In terms of model interpretation, maximum information preservation tried to interpret human nervous systems in terms of information storage and transmission in our neural systems.

Though this approach seemed to be a good starting point for interpretation, the definition and computation of mutual information inside cannot be necessarily and successfully used in the field of neural networks. Though many different types of methods have been developed so far [31]–[37], one of the main problems is the too-abstract property of mutual information, as well as other information measures applied to the neural networks. Thus, if we try to understand the inference mechanism in terms of information storage and transmission, we need to make this abstract property of mutual information more concrete and easily interpretable to be applicable to the interpretation of neural networks.

We should repeat again that one of the major problems with the Linsker-type maximum information preservation, when it is applied to neural networks, is how to measure information content in concrete components in neural networks. The present paper aims to realize the concept of mutual information in more concrete ways. In mutual information maximization in terms of neurons' information processing, two contradictory terms of entropy maximization and conditional entropy minimization exist. Using more concrete terms, individual neurons should respond very specifically to inputs, and at the same time, those neurons should respond very uniformly to any inputs on average or collectively. In living systems, neurons should be used as equally as possible on average, and at the same time, each neuron should respond as unequally as possible individually. The living systems must cope with many different types of new inputs and situations, and thus they need to use any resources inside as much possible, and at the same time, they need to know the properties of incoming inputs as much as possible. Thus, this information maximization principle states that the specific responses or larger information to targets, should be compensated for by the uniform use of components or smaller information from a collective point of view. In other words, it is necessary to have larger information, but this larger information should be distributed over as many components as possible.

Let us apply this knowledge of living systems from an information-theoretic view more concretely to neural networks. We suppose here that the information content is concretely measured in the selectivity of components such as neurons and connection weights. When we try to measure information content in neural networks, the selectivity of components should play an important role. Thus, information on inputs is stored and transmitted in terms of selectivity of components. When the selectivity of neurons toward inputs becomes higher, we should say that the neurons tend to have more information on inputs. Thus, we need to define the selectivity of components and how to control it.

The importance of selectivity in actual living systems, the interpretation methods, and in the field of improved generalization has been already pointed out. First, for actual living systems, it has been said that the selectivity should play an important role, discussed in the literature on neural sciences [38]–[44]. Second, as mentioned above, in the field of convolutional neural networks, to address the right to the explanation [9], a number of attempts have been made to interpret information processes in the networks. We think that the majority of methods have been based on the selectivity of components of neural networks [45]. Roughly speaking, the majority of interpretation methods have tried to determine which parts or components in multi-layered neural networks represent the distinctive or common features of input patterns. In other words, they have tried to determine which parts or components try to respond to inputs selectively. The interpretation of neural networks in this case corresponds to the determination of specific components for specific inputs. Third, though the paper does not deal with generalization, we should note that that generalization is directly related to the selectivity of components [45]–[50]. For example, when a neuron responds too specifically to some inputs, generalization performance cannot be improved, because it cannot respond appropriately to ambiguous input patterns. For improving generalization, we need to weaken the specific responses of neurons naturally.

This consideration leads us to say that the selectivity should be appropriately controlled in living systems for coping with uncertain conditions. Thus, we need to understand how this selectivity can be controlled in living systems and how they try to deal with uncertain conditions and unseen situations. The present paper aims to control the selectivity or selective information to explore how information should be stored and transmitted in neural networks.

One of the main hypotheses in this paper is to suppose that there is a variety of types of selectivity, and we should control those variants to cope with coming unseen inputs . In this paper, we suppose two types of selective information, namely, individual and collective selective information, or more simply, by eliminating the word "selective," individual and collective information. Individual information represents how much a component responds to an input specifically, namely, the information content an individual component has. On the contrary, collective information represent information pooled collectively by many neurons. Individual and collective information in neural networks are not necessarily in harmony with each other as is the case with human society. Thus, we need to make a compromise between individual and collective information in actual neural learning.

For simplicity's sake, we suppose here that individual information is naturally information on a specific and individual input, and it should be increased as much as possible. On the contrary, regarding collective information, it is supposed that we to collect as much information as possible on any inputs, and thus it is necessary for collective information to be as non-specific as possible to inputs. More technically, we suppose that individual information should be increased, while collective information should be decreased as much as possible. Then, we should try to examine what properties can be extracted when controlling two types of selective information.

As mentioned above, the maximum information principle by Linsker does not necessarily state that information should

 ISBN: 978-1-61208-882-2

be simply maximized. Information in the individual level can be increased under the condition that, in the collective level, information should be decreased. More concretely, information increase for individual components must be compensated for by information decrease from a collective point of view. In complex living systems, there are many different types of contradictions, where a contradiction can be resolved in a level, but in another level, the contradiction can be stronger. Thus, for the maximum information principle, we need to develop a composite information function by which information can be increased and decreased at the same time.

The present paper does not simply increase the selectivity of components. We try to control the selectivity of components individually and collectively. Then, we try to examine how information is stored and transmitted by changing the selectivity individually and collectively. We try to understand the main information processing mechanism in multi-layered neural networks by which we can explain the generation of individual inference mechanisms for different input patterns.

The paper has been organized as follows. In Section 2, we present how to compute two types of selective information, namely, individual and collective information. Then, we try to present how to modify connection weights by changing the ratio of individual to collective information. This method to modify connection weights is called "selective information-driven learning." Then, to understand the information flow in multi-layered neural networks, we show how to compress multi-layered neural networks step by step, namely, partial compression. In Section 3, we applied the method to the bankruptcy data set. In the experiments, we tried to show that the selective information could easily be controlled, and this control was directly related to which parts of information were obtained by multi-layered neural networks. By changing the ratio, we could produce compressed networks whose connection weights were quite similar to the original correlation coefficients between inputs and targets. In addition, we could extract and choose a few weights necessary only for prediction. The results show that the selective information control can generate more interpretable networks by which we can understand how information is processed in a multi-layered neural network.

II. Theory and Computational Methods

This section describes the selective information with individual and collective information. Then, we explain how to compress fully and partially networks into the simplest ones.

A. Selective Information Control

As discussed above, information contained in neural networks can be represented in terms of selectivity or selective information in their components. Thus, selective information is supposed to represent information content stored and transmitted in neural networks. Depending on given objectives, we need to control selective information appropriately, corresponding to the information control in neural networks. For example, too much selectivity of components is not good for responding to new inputs, but it is needed to specify the roles of components for the interpretation. In addition, for the efficient use of components, we need to treat those components as equally as possible. This means that we need to reduce the selectivity of components when we try to use the resources of components as much as possible. Then, the selectivity of components should be reduced, where each component should have equal importance collectively. Thus, we suppose that the selective information should be increased when components are treated individually. On the contrary, when they are treated collectively, the selectivity should be reduced.

For realizing this situation, we suppose two type of selective information. In one type of selective information, called "individual information," the information should be increased as much as possible. Each component should respond very specifically to inputs. On the contrary, in the other type of selective information, called "collective information," the information should be decreased as much as possible. All components should have the same importance; that is, all components should respond equally to inputs. By mixing these two types of information, we can control the selectivity of components. Naturally, it is not so easy to maximize and at the same time minimize the selective information in the same level. However, the selective information increase and decrease are performed in different levels, namely, individual and collective ones, and it is possible to make a compromise between two types of different selectivity control.

B. Individual Information

We explain here how to compute individual information. The individual information measures how much a neuron is connected with the corresponding one specifically. For this, we use a network architecture shown in Figure 1, in which the number of layers is seven, including the first input and the last seventh output layer. For simplicity's sake, we define them by connection weights between the second and third layer and neurons in the third layer in Figure 1. In our experiments, we tried to control the selective information only for hidden layers, because it is easy to do it in the hidden layers due to less input and output information.

First, the strength of connection weights can be computed by the absolute value

$$u_{jk}^{(2,3)} = \mid w_{jk}^{(2,3)} \mid \tag{1}$$

where $(2,3)$ denotes a transition from the second to the third layer. This strength denotes the strength of connecting a neuron with the other specific one. Because the strength is changeable, depending on different neurons, we normalize it by its maximum value

$$z_{jk}^{(2,3)} = \frac{u_{jk}^{(2,3)}}{\max_{k'} u_{jk'}^{(2,3)}} \tag{2}$$

By this normalized strength, we have individual information for the jth neuron

$$h_j^{(2,3)} = n_3 - \sum_{k=1}^{n_3} z_{jk}^{(2,3)} \tag{3}$$

where n_3 is the number of neurons in the third layer. This individual information increases when the number of stronger neurons decreases. Finally, when a neuron is connected only with one specific neuron, the selective information becomes

 ISBN: 978-1-61208-882-2

maximum $(n_3 - 1)$ in Figure 1(a). On the contrary, when the neuron is connected equally with all neurons, the selective information becomes minimum (zero) in Figure 1(b). When all connection weights are zero, the selective information should be zero by definition, because all connection weights have the same value of zero.

For the overall property of this selective information, by averaging it, we have the final individual information

$$h^{(2,3)} = \frac{1}{n_2} \sum_{j=1}^{n_2} h_j^{(2,3)} \tag{4}$$

This individual information roughly corresponds to conditional entropy in mutual information. However, this definition of information in terms of selectivity is more easily interpreted. Finally, we should note that, when we try to control the individual information, we should change weights according to the normalized value of z_{jk}.

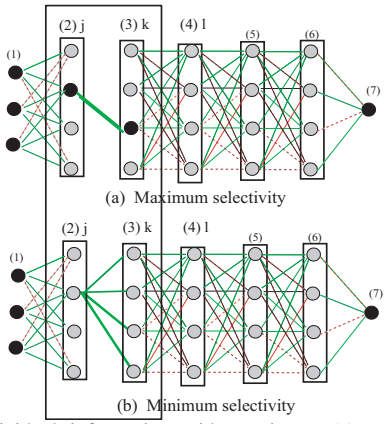

Fig. 1. Individual information with maximum (a) and minimum (b) selectivity.

C. Collective Information

We increase individual information and at the same time decrease the collective information. This is because we try to use all components as equally as possible for their efficient use, but we try to understand the meaning of each component as much as possible for interpretation. Simultaneous information maximization and minimization has difficulty for a compromise to be made between them. However, when those contradictory operations are performed in different levels, namely, individually and collectively , it is possible to do it.

Then, the collective information should denote information, independently of specific neurons, as well as connection weights. Thus, as shown in Figure 2(a) for the collective information, we sum the strength of connection weights

$$u_k^{(2,3)} = \sum_{j=1}^{n_2} u_{jk}^{(2,3)} \tag{5}$$

where n_2 denotes the number of neurons in the second layer. The normalized strength is computed by

$$v_k^{(2,3)} = \frac{u_k^{(2,3)}}{\max_{k'} u_{k'}^{(2,3)}} \tag{6}$$

Using this normalized strength, collective information is defined by

$$g^{(3)} = n_3 - \sum_{k=1}^{n_3} v_k^{(2,3)} \tag{7}$$

When this collective information, all connection weights to the corresponding neurons in the third layer becomes stronger, as shown in Figure 2(a). On the contrary, when the information decreases, all connection weights to the neurons become equal, as shown in Figure 2(b). As mentioned above, all connection weights happen to be very strong when the information is maximized, as shown in Figure 2(b). Thus, we need to reduce the strength of weights as much as possible for the minimum information states shown in Figure 2(c). Collective information minimization is a good candidate for information minimization, because this minimization aims to make all connection weights equal only collectively, meaning that some weights may be relatively stronger. This property can be good at compromising between individual and collective information.

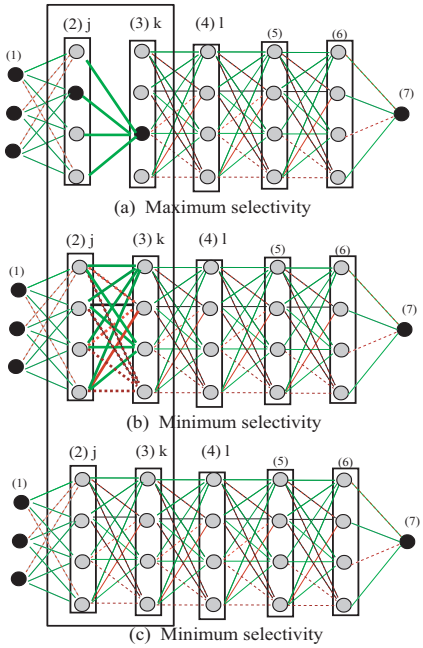

Fig. 2. Collective information with maximum selective information (a) and minimum selective information (b) and (c).

D. Selective Information-Driven Learning

We must control two types of selective information by actually changing the weights. However, to control the selective information, all we have to do is to change weights by the normalized strength of connection weights z and v. For this purpose, we introduce a composite measure, combining the normalized strength of weights

$$d_{jk}^{(2,3)} = \alpha z_{jk}^{(2,3)} + \bar{\alpha} \bar{v}_k^{(3)} \tag{8}$$

where parameter α ranges between zero and one, $\bar{\alpha} = 1 - \alpha$, and $\bar{v} = 1 - v$. When the parameter α increases, the effect of individual information increases. On the contrary, when the parameter α decreases and $\bar{\alpha}$ increases, the effect of collective information increases. When the effect of $\bar{\alpha}$ increases, the strongest weights are forced to be smaller, keeping smaller

 ISBN: 978-1-61208-882-2

ones relatively the same. Eventually, all connection weights become equal and smaller.

Then, for the connection weights from the second to the third layer in the tth learning step, the weights are changed simply by

$$w_{jk}^{(2,3)}(t+1) = d_{jk}^{(2,3)}(t)\ w_{jk}^{(2,3)}(t) \tag{9}$$

When the composite measure d is applied, weights should be updated with the normal error minimization method to assimilate the effect of the composite measure.

E. Partial Compression

To examine more carefully the information flow by the selective information control, we try to see how connection weights are changed when going through multiple hidden layers. For this purpose, we introduce partial compression, where an original multi-layered neural network is gradually compressed into the simplest one without hidden layers for interpretation. For simplicity's sake, the number of neurons in any hidden layers was the same. This assumption of an equal number of neurons in hidden layers does not necessarily mean that we do not consider a different number of neurons in hidden layers. The reduction in the number of neurons can be actually realized by suppressing the number of connection weights by the present method.

Figure 3 shows how to compress a multi-layered neural network step by step. In the first partial compression, as shown in Figure 3(a), we immediately compress the input and output layers, skipping all hidden layers,

$$w_{iq}^{(1,7)} = \sum_{q=1}^{n_6} w_{iq}^{(1,2)} w_{qr}^{(6,7)} \tag{10}$$

This shows only information contained in the input and output layers.

Then, we suppose that two connection weights from the first to the second layer, represented by (1,2), and between the second and the third layer, represented by (2,3), are combined into

$$w_{ik}^{(1,2,3)} = \sum_{j=1}^{n_2} w_{ij}^{(1,2)} w_{jk}^{(2,3)} \tag{11}$$

where (1,2,3) represents compression that is performed up to the second layer. In the second compression in Figure 3(b), we combine this compressed weight with the final output layer

$$w_{ir}^{(1,3,7)} = \sum_{q=1}^{n_6} w_{iq}^{(1,2,3)} w_{qr}^{(6,7)} \tag{12}$$

where (1,3,7) denotes that compression is performed up to the third layer. We compress the remaining connection weights in the same way in Figure 3(c) and (d), and finally, we can compress all layers into the simplest ones in Figure 3(e)

$$w_{ir}^{(1,6,7)} = \sum_{q=1}^{n_6} w_{iq}^{(1,6,6)} w_{qr}^{(6,7)} \tag{13}$$

where $(1, 6, 7)$ shows that compression is performed up to the sixth layer, namely, full compression. Those compressed weights aim to represent the main characteristics of overall connection weights. The compressed weights can be computed by multiplying connection weights of all routes from an input to the corresponding output.

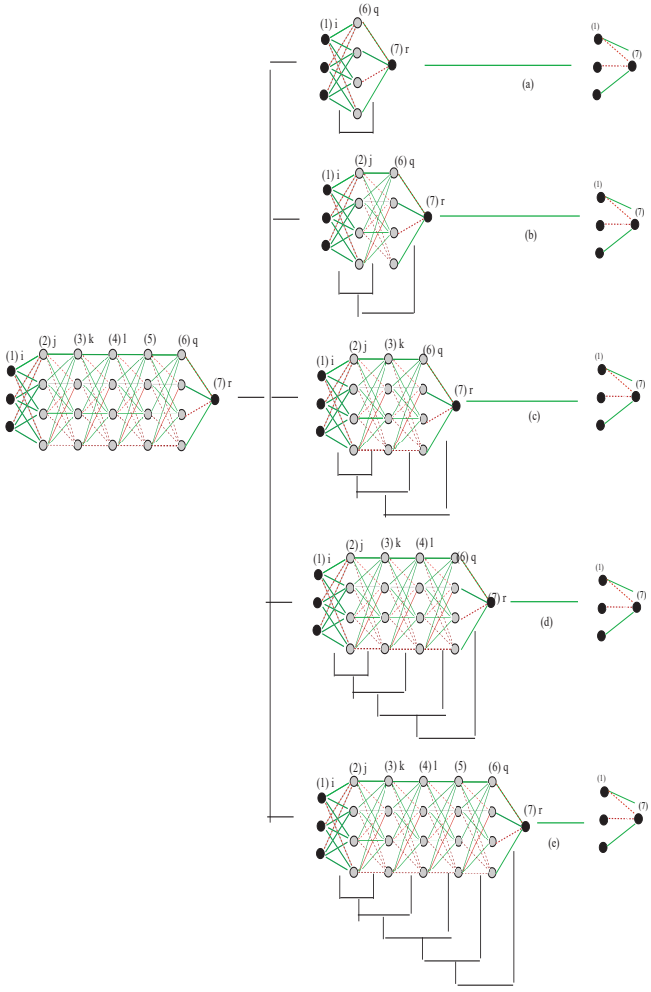

Fig. 3. Network architecture with seven layers, including five hidden layers (a) to be compressed step by step into the simplest ones (b)-(e).

III. Results and Discussion

This section present the experimental results applied to the bankruptcy data set, in which we tried to show that individual and collective information could be controlled to capture input or output information. In addition, behind complicated connection weights, we could find very simple, independent, and individual relations between inputs and outputs.

A. Results of Bankruptcy Data Set

1) Experimental Setting: The experiment aimed to predict bankruptcy by six qualitative input variables: industrial risk, management risk, financial flexibility, credibility, and operating risk [51]. As shown in Figure 4, the number of input, hidden, and output neurons was 6, ten, and one, and the number of hidden layers was ten. We used the partial compression in which compression was performed step by step by multiplying connection weights in higher hidden layers as shown on the lower side of the figure in Figure 4. The number of input patterns was 250, where 70 percent and the remaining 30

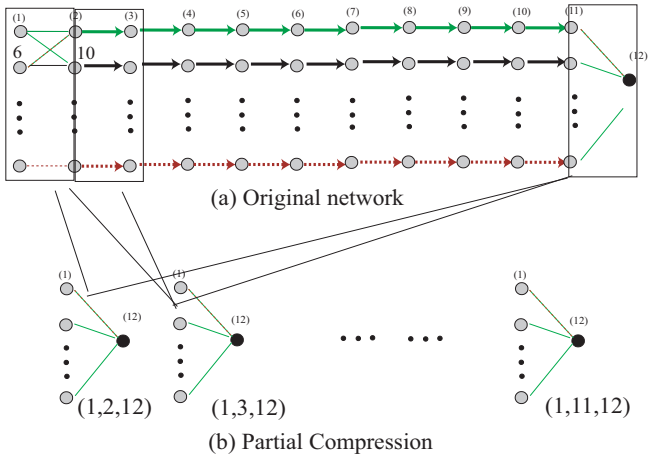

Fig. 4. Network architecture (a) and a process of partial compression (b) for the bankruptcy data set. The notation (1,2,12) denotes that compression is applied up to the second layer.

percent of the data set was for training and testing. The generalization performance was almost perfect by the present method, as well as the other conventional methods such as logistic regression and random forest. However, connection weights and other importance measures were different by different methods. Thus, this is a good benchmark data set for comparing the final representations by different methods. We used the scikit-learn neural network package with almost all parameters set to default values, except for the number of epochs and tangent-hyperbolic activation function, for making the reproduction of the present results as easy as possible. In addition, the effect of selective information is forced to be applied many times (up to 5 times), in direct proportion to the number of learning epochs. This property tended to make connection weights extremely large or small due to the repeated assimilation procedures. Thus, we reduced the effect of selective information as follows:

$$d_{jk}^{(2,3)} = \theta \left[\alpha z_{jk}^{(2,3)} + \bar{\alpha} \bar{v}_k^{(3)} \right]^{\beta} \tag{14}$$

Newly added parameters θ and β were used only for reducing the excessive effects of selective information by repeating the assimilation processes (up to 5 times). The actual values of the parameter θ and β were 0.99 and 0.9, respectively.

2) Selective Information Control: In the first place, we try to show that the present method can decrease collective information and at the same time increase individual information by changing the parameter α. Figure 5 shows collective and individual information and the ratio of individual to collective information for the last hidden layer, namely, from the tenth to the eleventh layer for the bankruptcy data set. We chose the most typical hidden layer, where selective information could be controlled the most explicitly by the present method. As shown in Figure 5(a1), collective information increased very slightly, and individual information increased immediately in the first place and remained almost the same in the later stages of the learning steps. Thus, the ratio of individual to collective information stayed almost the same throughout the entire learning steps. This can be explained by the fact that, when the parameter was one, only individual information was forced to be increased. When the parameter decreased from 0.9 (b) to 0.7 (d), individual information was forced to increase

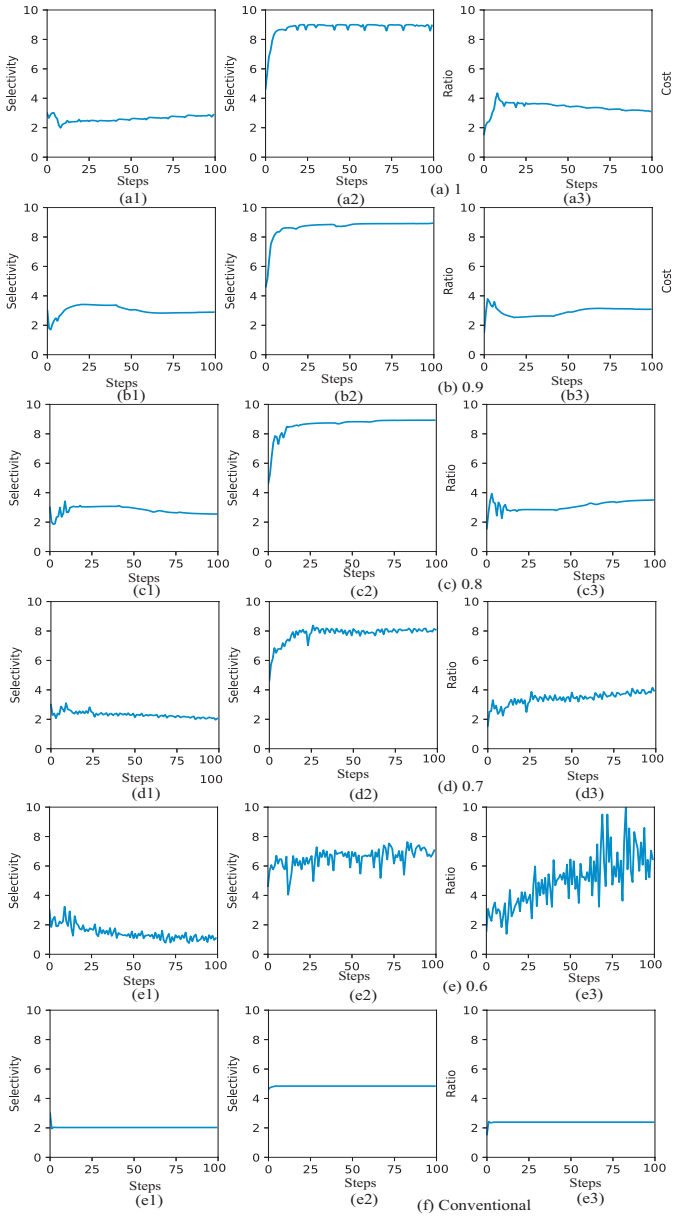

Fig. 5. Collective information (1), individual information (2), and the ratio of individual to collective information (3) when the parameter decreased from 1 (a) to 0.6 (e), and the conventional method (f) for the bankruptcy data set.

more slowly. Collective information gradually decreased, and the ratio of collective to individual information increased gradually. In particular, when the parameter was 0.6 in Figure 5(e3), the ratios increased considerably when the learning steps increased. However, we could see some fluctuations, showing difficulty in controlling two types of information. Note that, when the parameter was increased further, we could not obtain stable results. Finally, without information control, no changes in selective information and its ratios could be seen in Figure 5(f). These results show that the present method can control two types of selective information, where collective information can be decreased, and at the same time, individual information can be increased, though some difficulty could be seen when the parameter was forced to be smaller and we must compromise between individual and collective information.

 ISBN: 978-1-61208-882-2

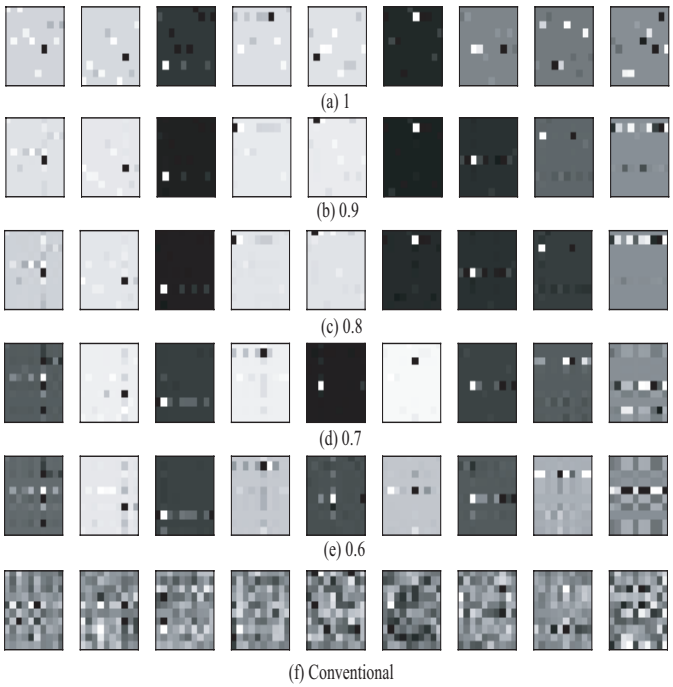

Fig. 6. Weights when the parameter decreased from 1 (a) to 0.6 (e) and by using the conventional method (f) for the bankruptcy data set.

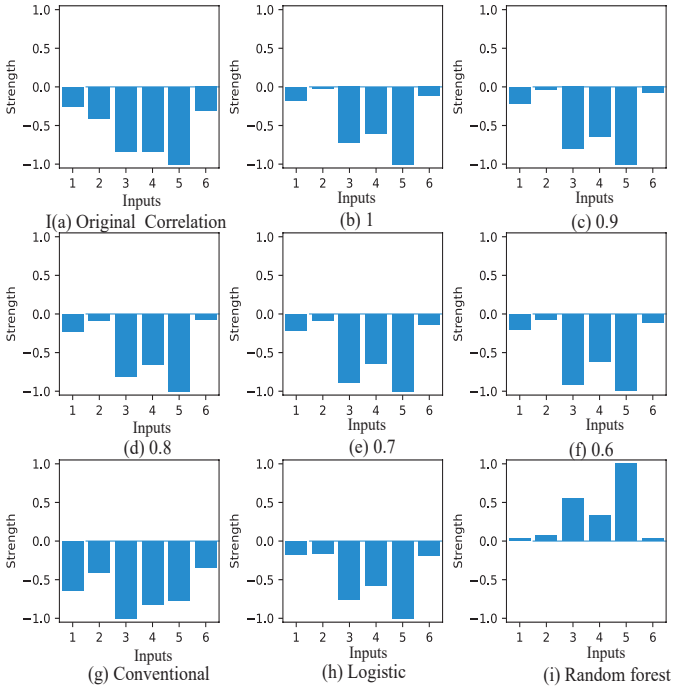

Fig. 7. Correlation coefficients between inputs and targets in the original data set (a) and compressed weights when the parameter α decreased from 1 (b) to 0.6 (f) and by the conventional method without selective information control (g), and the regression coefficients by the logistic regression analysis (h) and prediction importance (i) by the random forest method for the bankruptcy data set.

Figure 6 shows connection weights for all hidden layers when the parameter decreased from 1(a) to 0.6(e). When the parameter was one in Figure 6(a), many strong connection weights were scattered. When the parameter decreased from 0.9 in Figure 6(b) to 0.7 in Figure 6(d), a neuron in the preceding layer tended to be connected with all neurons in the subsequent layer. In particular, the final connection weights from the tenth to the eleventh layer, located in the rightmost column, shows the most typical state. This tendency tended to prevail for all layers when the parameter decreased to 0.6 in Figure 6(e). This means that, when collective information is forced to decrease, a specific neuron tends to be connected with all neurons in the subsequent layer. Collective information minimization can be realized by connecting a neuron with as many different neurons as possible. Finally, when we did not use selective information, little regularity could be seen over connection weights in Figure 6(f). However, we could see the same tendency, that neurons in the precedent layers were connected with ones in the subsequent layers. The selective information method seems to enhance this tendency.

3) Compressed Weights: We show here that the compressed weights from the original multi-layered neural networks were very close to the original correlation coefficients between inputs and targets of the data set, meaning that the method could disentangle connection weights to get individual and independent relations between inputs and outputs.

Figure 7(a) shows the correlation coefficients between inputs and targets of the original data set. Figures 7 (b) to (f) show compressed weights when the parameter decreased from 1 to 0.6. In addition, Figure 7(h) shows the regression coefficients of the logistic regression analysis. Those compressed weights and regression coefficients were quite similar to each other, and they were close to the original correlation coefficients between inputs and outputs in Figure 7(a). On the contrary, the conventional method without the selective information produced compressed weights different from the original correlations and regression coefficients in Figure 7(g). This means that the selective information control can be used to disentangle connection weights and to produce individual and independent relations between inputs and outputs. Finally, Figure 7(i) shows the prediction importance by the random forest method, where the importance was all positive, because the method could not deal with the negative values.

4) Partially Compressed Weights: Then, we tried to partially compress connection weights to examine how neural networks tried to extract information when the hidden layer increased. The principal finding was that individual information tended to deal with information from outputs, while collective information tended to focus on information from inputs.

Figure 8(a) shows compressed weights for all layers when the parameter was one and only individual information was forced to be increased. Weights were partially and step by step compressed from the top- left to the bottom-right box. The top-left compressed weights were ones obtained by combining only input and output layers, and the bottom-right weights were obtained by compressed all connection weights of all layers, namely, full compression. As can be seen in the figures, partially compressed weights were small when we compressed weights up to the sixth layer, when the compressed weights became slightly stronger. Finally, we could obtain the final compressed weights by the full compression (bottom-right). This means that neural networks cannot extract necessary information up to the final layer, which suggests that the individual information is dependent on the acquisition of output information.

 ISBN: 978-1-61208-882-2

Figure 8(b) shows partially compressed weights when the parameter was set to zero and only collective information was used. Note that, when the parameter was set to zero, the neural networks could not finish the learning, producing large errors for the outputs. Thus, the results were used to show the effects of collective information minimization as clearly as possible. As can be seen in the figure, only when connection weights in the input and output layers were compressed (top-left), we could obtain strong connection weights, while in all the other cases, the compressed weights were very small. As was mentioned, the learning failed, because information from inputs could not be transmitted to the final layer.

Figure 8(c) shows partially compressed weights by the conventional method without selective information. As can be seen in the figure, only when all connection weights in all layers were compressed (bottom right) were strong compressed weights obtained. This seems to us that the conventional method focused on information from outputs. As shown in Figure 8(a), individual information maximization also showed the same tendency of focusing toward output information.

These results show that the conventional method, as well as individual information can produce the final compressed weights only when all information goes through all hidden layers. On the other hand, the collective information has a property to detect some information in the early stages of multi-layers. From these experimental results, we can infer that we can obtain different types of internal representations by focusing on input or output information.

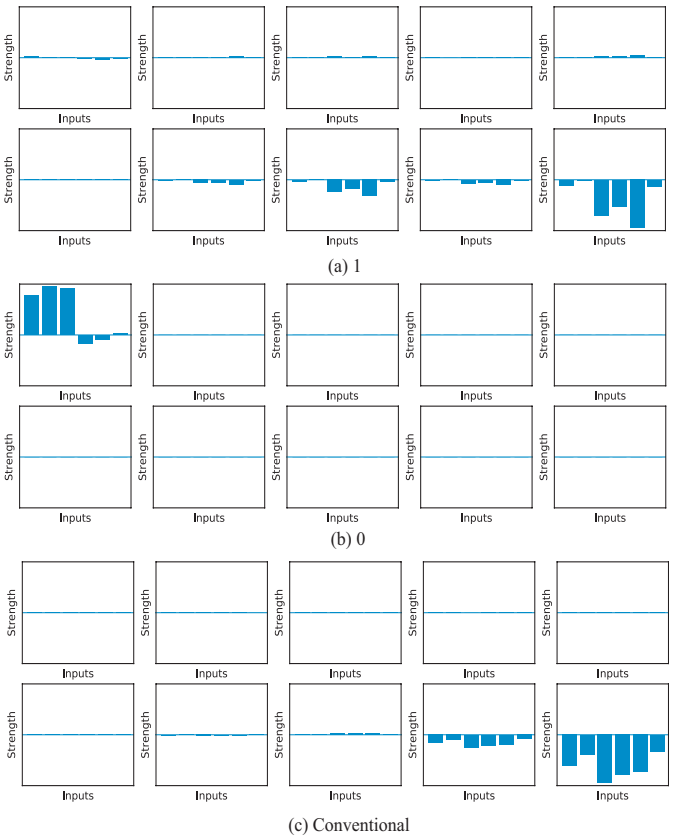

Fig. 8. Partially compressed weights when the parameter was 1 (a) and 0 (b), and the conventional method without selective information (c) for the bankruptcy data set.

IV. Conclusion and Future Work

The present paper aimed to propose a new type of information-theoretic method to control the selectivity of components of neural networks. To interpret the process of information storing and transmission of neural networks, we need to control the selectivity of components, or selective information. Only by controlling the selective information can we interpret the information processing of neural networks and thus interpret the main inference mechanism of neural networks. We prepared two types of selective information: individual and collective information. By the ratio of the two types of selective information, we tried to explore the main information processing mechanism. The method was applied to the bankruptcy business data sets. The experimental results showed that individual and collective information could be controlled by the present method. With this control, we could see how neural networks try to capture input information or output information differently. In addition, the results showed that, behind seemingly complicated representations in multi-layered neural networks, very simple, individual, and independent relations could be observed. It can be expected that the complicated representations in the surface level can be transformed from those simple basic representations. We should say that it is possible to transform complicated neural networks into the simple ones, whose basic structure can be easily interpreted, and the structure can be easily transformed to produce a variety of surface and complicated representations.

Finally, this paper was concerned with the interpretation, but it is better to unify the problem of interpretation with improved generalization performance. Thus, we need to propose a method to interpret the inference mechanism, followed by improved generalization.

References

[1] A. Hart and J. Wyatt, "Evaluating black-boxes as medical decision aids: issues arising from a study of neural networks," *Medical informatics*, vol. 15, no. 3, pp. 229–236, 1990.

[2] M. Spining, J. Darsey, B. Sumpter, and D. Nold, "Opening up the black box of artificial neural networks," *Journal of chemical education*, vol. 71, no. 5, p. 406, 1994.

[3] J. M. Benítez, J. L. Castro, and I. Requena, "Are artificial neural networks black boxes?," *IEEE Transactions on Neural Networks*, vol. 8, no. 5, pp. 1156–1164, 1997.

[4] J. D. Olden and D. A. Jackson, "Illuminating the black box: a randomization approach for understanding variable contributions in artificial neural networks," *Ecological modelling*, vol. 154, no. 1-2, pp. 135–150, 2002.

[5] F. Qiu and J. Jensen, "Opening the black box of neural networks for remote sensing image classification," *International Journal of Remote Sensing*, vol. 25, no. 9, pp. 1749–1768, 2004.

[6] G. Bologna, "Is it worth generating rules from neural network ensembles?," *Journal of Applied Logic*, vol. 2, no. 3, pp. 325–348, 2004.

[7] C. Rudin, "Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead," *Nature Machine Intelligence*, vol. 1, no. 5, pp. 206–215, 2019.

[8] M. Sendak et al., "The human body is a black box" supporting clinical decision-making with deep learning," in *Proceedings of the 2020 Conference on Fairness, Accountability, and Transparency*, pp. 99–109, 2020.

[9] B. Goodman and S. Flaxman, "European union regulations on algorithmic decision-making and a right to explanation," *arXiv preprint arXiv:1606.08813*, 2016.

[10] K. R. Varshney and H. Alemzadeh, "On the safety of machine learning: Cyber-physical systems, decision sciences, and data products," *Big data*, vol. 5, no. 3, pp. 246–255, 2017.

[11] D. E. Rumelhart, G. E. Hinton, and R. Williams, "Learning internal representations by error propagation," in *Parallel Distributed Processing* (D. E. Rumelhart and G. E. H. et al., eds.), vol. 1, pp. 318–362, Cambridge: MIT Press, 1986.

[12] D. E. Rumelhart and D. Zipser, "Feature discovery by competitive learning," *Cognitive Science*, vol. 9, pp. 75–112, 1985.

[13] D. E. Rumelhart and J. L. McClelland, "On learning the past tenses of English verbs," in *Parallel Distributed Processing* (D. E. Rumelhart, G. E. Hinton, and R. J. Williams, eds.), vol. 2, pp. 216–271, Cambrige: MIT Press, 1986.

[14] D. Rumelhart and J. M. et al., *Parallel Distributed Processing*, vol. 1. MA: MIT Press, 1986.

[15] K. Fukushima, "Cognitron: a self-organizing multi-layered neural network," *Biological Cybernetics*, vol. 20, pp. 121–136, 1975.

[16] K. Fukushima, "Neocognitron: a hierarchical neural network capable of visual pattern recognition," *Biological Cybernetics*, vol. 20, pp. 121–136, 1975.

[17] K. Fukushima, "Neocognitron: a neural network model for a mechanism of visual pattern recognition," *IEEE Transactions on Systems, Man, and Cyvernetics*, vol. 13, pp. 826–834, 1983.

[18] D. H. Hubel and T. N. Wisel, "Receptive fields, binocular interaction and functional architecture in cat's visual cortex," *Journal of Physiology*, vol. 160, pp. 106–154, 1962.

[19] A. Nguyen, J. Yosinski, and J. Clune, "Understanding neural networks via feature visualization: A survey," in *Explainable AI: Interpreting, Explaining and Visualizing Deep Learning*, pp. 55–76, Springer, 2019.

[20] G. Montavon, W. Samek, and K.-R. Müller, "Methods for interpreting and understanding deep neural networks," *Digital Signal Processing*, vol. 73, pp. 1–15, 2018.

[21] D. Erhan, Y. Bengio, A. Courville, and P. Vincent, "Visualizing higher-layer features of a deep network," *University of Montreal*, vol. 1341, 2009.

[22] K. Simonyan, A. Vedaldi, and A. Zisserman, "Deep inside convolutional networks: Visualising image classification models and saliency maps," *arXiv preprint arXiv:1312.6034*, 2013.

[23] S. Bach, et al., "On pixel-wise explanations for non-linear classifier decisions by layer-wise relevance propagation," *PloS one*, vol. 10, no. 7, p. e0130140, 2015.

[24] F. Arbabzadah, G. Montavon, K.-R. Müller, and W. Samek, "Identifying individual facial expressions by deconstructing a neural network," in *German Conference on Pattern Recognition*, pp. 344–354, Springer, 2016.

[25] R. R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh, and D. Batra, "Grad-cam: Visual explanations from deep networks via gradient-based localization," in *Proceedings of the IEEE international conference on computer vision*, pp. 618–626, 2017.

[26] R. Fu, Q. Hu, X. Dong, Y. Guo, Y. Gao, and B. Li, "Axiom-based grad-cam: Towards accurate visualization and explanation of cnns," *arXiv preprint arXiv:2008.02312*, 2020.

[27] R. Linsker, "Self-organization in a perceptual network," *Computer*, vol. 21, no. 3, pp. 105–117, 1988.

[28] R. Linsker, "How to generate ordered maps by maximizing the mutual information between input and output signals," *Neural computation*, vol. 1, no. 3, pp. 402–411, 1989.

[29] R. Linsker, "Local synaptic learning rules suffice to maximize mutual information in a linear network," *Neural Computation*, vol. 4, no. 5, pp. 691–702, 1992.

[30] R. Linsker, "Improved local learning rule for information maximization and related applications," *Neural networks*, vol. 18, no. 3, pp. 261–265, 2005.

[31] S. Becker, "Mutual information maximization: models of cortical self-organization," *Network: Computation in Neural Systems*, vol. 7, pp. 7–31, 1996.

[32] K. Torkkola, "Nonlinear feature transform using maximum mutual information," in *Proceedings of International Joint Conference on Neural Networks*, pp. 2756–2761, 2001.

[33] K. Torkkola, "Feature extraction by non-parametric mutual information maximization," *Journal of Machine Learning Research*, vol. 3, pp. 1415–1438, 2003.

[34] J. M. Leiva-Murillo and A. Artés-Rodríguez, "Maximization of mutual information for supervised linear feature extraction," *Neural Networks, IEEE Transactions on*, vol. 18, no. 5, pp. 1433–1441, 2007.

[35] M. M. Van Hulle, "The formation of topographic maps that maximize the average mutual information of the output responses to noiseless input signals," *Neural Computation*, vol. 9, no. 3, pp. 595–606, 1997.

[36] J. C. Principe, D. Xu, and J. Fisher, "Information theoretic learning," *Unsupervised adaptive filtering*, vol. 1, pp. 265–319, 2000.

[37] J. C. Principe, *Information theoretic learning: Renyi's entropy and kernel perspectives*. Springer Science & Business Media, 2010.

[38] E. L. Bienenstock, L. N. Cooper, and P. W. Munro, "Theory for the development of neuron selectivity," *Journal of Neuroscience*, vol. 2, pp. 32–48, 1982.

[39] A. Schoups, R. Vogels, N. Qian, and G. Orban, "Practising orientation identification improves orientation coding in v1 neurons," *Nature*, vol. 412, no. 6846, pp. 549–553, 2001.

[40] L. E. White, D. M. Coppola, and D. Fitzpatrick, "The contribution of sensory experience to the maturation of orientation selectivity in ferret visual cortex," *Nature*, vol. 411, no. 6841, pp. 1049–1052, 2001.

[41] H. Ko, S. B. Hofer, B. Pichler, K. A. Buchanan, P. J. Sjöström, and T. D. Mrsic-Flogel, "Functional specificity of local synaptic connections in neocortical networks," *Nature*, vol. 473, no. 7345, pp. 87–91, 2011.

[42] J. F. Jehee, S. Ling, J. D. Swisher, R. S. van Bergen, and F. Tong, "Perceptual learning selectively refines orientation representations in early visual cortex," *Journal of Neuroscience*, vol. 32, no. 47, pp. 16747–16753, 2012.

[43] M. V. Peelen and P. Downing, "Category selectivity in human visual cortex," 2020.

[44] B. J. Bongers, A. P. IJzerman, and G. J. Van Westen, "Proteochemometrics–recent developments in bioactivity and selectivity modeling," *Drug Discovery Today: Technologies*, 2020.

[45] M. L. Leavitt and A. Morcos, "Selectivity considered harmful: evaluating the causal impact of class selectivity in dnns," *arXiv preprint arXiv:2003.01262*, 2020.

[46] A. S. Morcos, D. G. Barrett, M. Botvinick, and N. C. Rabinowitz, "On the importance of single directions for generalization," 2018.

[47] I. Rafegas, M. Vanrell, L. A. Alexandre, and G. Arias, "Understanding trained cnns by indexing neuron selectivity," *Pattern Recognition Letters*, vol. 136, pp. 318–325, 2020.

[48] J. Ukita, "Causal importance of low-level feature selectivity for generalization in image recognition," *Neural Networks*, vol. 125, pp. 185–193, 2020.

[49] W. J. Johnston, S. E. Palmer, and D. J. Freedman, "Nonlinear mixed selectivity supports reliable neural computation," *PLoS computational biology*, vol. 16, no. 2, p. e1007544, 2020.

[50] M. L. Leavitt and A. S. Morcos, "On the relationship between class selectivity, dimensionality, and robustness," *arXiv preprint arXiv:2007.04440*, 2020.

[51] M.-J. Kim and I. Han, "The discovery of experts' decision rules from qualitative bankruptcy data using genetic algorithms," *Expert Systems with Applications*, vol. 25, no. 4, pp. 637–646, 2003.

 ISBN: 978-1-61208-882-2

A Hybrid Model to Improve Occluded Facial Expressions Prediction in the Wild during Conversational Head Movements

Arvind K. Bansal
Department of Computer Science
Kent State University
Kent, OH, USA
email: arvind@cs.kent.edu

Mehdi Ghayoumi
eCornell and Department of Computer Science
Cornell University and University of San Diego
San Diego, CA, USA
email: mg948@cornell.edu

Abstract— **Human emotion prediction is an important aspect of conversational interactions in social robotics. Conversational interactions involve a combination of dialogs, facial expression, speech modulation, pose analysis, head gestures, and hand gestures in varying lighting conditions and noisy environment involving multi-party interaction. Head motions during conversational gestures, multi-agent conversations and varying lighting conditions cause occlusion of the facial feature-points. Popular Convolution Neural Network (CNN) based predictions of facial expressions degrade significantly due to occluded feature-points during extreme head-movements during conversational gestures and multi-agent interaction in real-world scenarios. In this research, facial symmetry is exploited to reduce the loss of discriminatory feature-point information during conversational head rotations. CNN-based model is augmented with a new rotation invariant symmetry-based geometric modeling. The proposed geometric model corresponds to Facial Action Units (FAU) for facial expressions. Experimental data show hybrid model comprising a CNN-based model and the proposed geometric model outperforms the CNN-based model by 8%-20%, depending upon the type of facial-expression, beyond partial head rotations.**

Keywords-Artificial Intelligence; conversation; emotion analysis; facial expression analysis; facial occlusion; facial symmetry; head movement; multimedia.

I. INTRODUCTION

Due to an aging population in the developed world and limited workforce [1], there is a growing need of social robotics for elderly care and healthcare [2]. To show empathy, interact, and converse with humans, social robots need to understand human emotions and pain [3], [4] in the wild i.e. emotions are derived from the living systems in real-time scenarios, such as attending elderly patients or helping a frustrated elderly person in a home setting [5], [6].

Predicting emotions in the wild is complex and requires multimodal multimedia analysis involving dialogs [7], voice-modulation (including timed silence) [8], gestures (including postures, gaze, conversational head and hand gestures, and haptic gestures) [9], facial expressions [10]-[13], pain, and tears. Many desirable human-robot interactions, such as conversational gestures, including human warmth and affection, frustration, irritation, encouragement, impatience and pain shown by a combination of voice-modulation, speech-phrases, gestures, facial expressions are yet to be achieved. Compared to emotions exhibited in dialogs, utterances and gestures, facial expressions are exhibited more involuntarily [4], [10].

Multimedia analysis of facial expressions requires a sequence of video frames, dimension reduction, intelligent image analysis, and analysis of the intensity of facial expressions. In cognitive psychology, two approaches are used to study facial expressions: basic six emotions (anger, disgust, fear, happiness, sadness, and surprise) popularized by Ekman and others [14]; Valence-based Plutchik's wheel of emotion that relates positive and negative emotion classes in multiple intensity levels [15]. Computational analysis is currently limited to recognizing six basic emotions [4], [10], [11] due to tractability of the underlying problem and explicit correspondence of basic emotions to facial muscles modeled by facial action units [14].

Previous studies are mostly limited to the frontal facial view [11] or static aligned poses [16] using curated databases [17]-[20] showing nonoccluded pure facial expressions in proper lighting conditions. In recent years, many researchers have suggested techniques to handle information loss caused by partial occlusion due to external face-obstructing objects, such as eye-glasses, hats, scarfs, and medical masks; hand gestures; hair and mustaches; ambient lighting conditions [21]-[29]. These schemes are based on reconstruction of small patches of a partially occluded face using nonoccluded (or global) facial texture. None of these techniques are suited for extreme loss of discriminatory feature-points during extreme head-rotations in argumentation, denial or multi-party interactions where a significant part of face is occluded for a longer period.

In real-life scenarios, the face continuously moves during a conversation [9], [30], [31] based upon 1) conversational gestures, such as argumentation, interrogation and denial [9]; 2) intensity of emotion [15]; 3) multi-party interactions; changing ambient lighting conditions and shadows with head-movements during conversational gestures. Head rotations stochastically occlude feature-points causing information loss hindering accurate facial-expression classification.

Experiments with CNN-based model [13], as described in section 5, show that facial expression prediction drops by 10-

 ISBN: 978-1-61208-882-2

20% for partial occlusion (less than 45° rotation) and by 30-50% beyond 45° rotation.

Recent augmentation of CNN-based modeling with Long Short Term Memory (LSTM) and transfer learning improves temporal context and maps real-time movement to the nearest alignment of static CNN model to improve the prediction [31] during head-movement. However, they do not handle extreme information loss beyond partial occlusion and do not exploit facial symmetry.

CNN-based models need to be augmented with temporal contexts and restore occluded discriminatory feature-points for beyond the partial occlusion in conversational head-gestures, such as emotional disagreement, interrogation, argumentation or denial; multi-party interaction that involves significant occlusion of one part of the face. Luckily, even during extreme head-rotation, only one side of the face is occluded, and facial symmetry can be used to reconstruct the occluded discriminatory feature-points knowing the coordinates of their counterparts on the nonoccluded side.

This research improves facial-expression analysis for face under motion by utilizing facial symmetry [32] along the vertical major axis. Facial symmetry has been used to estimate the coordinates of missing discriminatory feature-points using their nonoccluded counterparts [33], [34]. Prediction is based upon 1) inherent symmetry of the face around the vertical axis of the face; 2) noted differences between the symmetrical points and the actual geometric feature-points from the previous frames.

The proposed hybrid model augments the CNN-based model [13] with a symmetry-based geometric model proposed in this paper. The hybrid model uses CNN-based prediction for the nonoccluded or partially occluded space and the symmetry-based geometric model beyond partially occluded space. The proposed geometric model provides motion continuity and temporal context to the CNN model for selecting the nearest static alignment.

The major contributions in this research are:

1. Development of a symmetry-based geometric model corresponding to Facial Action Units (FAUs) to recover discriminative feature-points during conversational head-rotations in real-time scenarios;
2. Augmentation of the CNN-based model with the proposed symmetry-based geometric model to improve the temporal context and the facial expression prediction beyond the partial occlusion.

The overall roadmap of this paper is: Section 2 describes the related work. Section 3 describes background concepts about facial features. Section 4 describes the proposed symmetry-based geometric model. Section 5 describes the implementation and experimental results. Section 6 concludes the paper.

II. RELATED WORK

Related work can be classified as: 1) handling occlusion for improper lighting conditions, hand-gestures and external objects [24]-[32]; 2) analyzing emotions in the wild [6]; 3) a combination of CNN, LSTM and transfer learning to map continuous motion to the corresponding CNN [31].

To handle the occlusion caused by external objects, researchers have used fixed pose alignments [16], hybrid models training on occluded and nonoccluded samples and using nonoccluded features-space as a guidance to predict texture of occluded patches [24], combination of sparse representation and maximum likelihood estimation [25], a combination of Gabor filter and local binary pattern to derive the texture of occluded patch [26], deep structure recognition [27], a combination of feature histogram, dimension reduction and support vector machine [28], Gabor filter and co-occurrence matrices [29], combination of global and local textures with CNN and attention [30], use of LSTM auto-encoders [31], and Bayesian networks [32].

The above schemes combine information from previous images or texture-pattern from nonoccluded space, and dynamic weighting of texture-patterns to reconstruct occluded patches using well curated datasets [17]-[20]. These schemes do not analyze occluded facial expressions in the wild during conversational head-motion.

Zong et al. use a combination of transduction transfer learning and linear discriminant analysis to map the trained data using curated dataset to the data in the wild [13]. However, the scheme does not: 1) handle conversational head movements and the resulting occlusion; 2) does not use symmetry to recover occluded feature-points.

T-H. S. Li et al. integrate CNN with LSTM to provide the temporal context [23] required for analyzing facial expression during head rotation. They use transfer learning to map a position to the corresponding static alignment of CNN for improved accuracy. The scheme is limited by the number of fixed domains for transfer learning and does not exploit symmetry. Besides, LSTM cannot estimate the coordinates of the occluded feature-points explicitly.

Compared to other schemes, the proposed geometric model significantly exploits facial symmetry [33] - [35] to recover occluded feature-points during extreme head rotations. The correspondence of the line-segments joining discriminatory feature-points to Facial Action Units (FAUs) relates the proposed hybrid model with Facial Action Coding System (FACS) based analysis and CNN-based analysis. In addition, the changes in line-segment ratios with head-movements provide temporal context even under extreme head-rotations. In our scheme, the availability of discriminative feature-points supports multimodal analysis of head-gestures and provides explanation capability.

III. BACKGROUND

A face has two types of feature-points: *fixed points* and *active points*. *Fixed points* act as a reference, and *active-points* move during facial-expressions, altering x and z-coordinates of feature-points [10]. Figure 1 illustrates various feature points.

A face has six major *fixed points*: two ends of the left and right eyes; bottom of a nose; middle point between two eye-brows above the nose-tip. There are 14 major active points: 1) three points on each brow; 2) two middle points of lips; 3) two endpoints of the mouth; 4) two middle points in each eye.

 ISBN: 978-1-61208-882-2

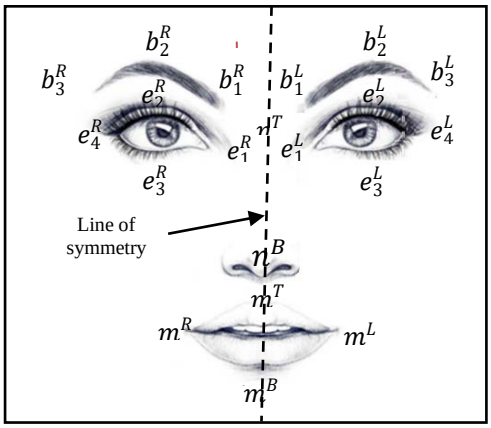

Figure 1. Facial feature points with symmetry

Feature-points' denotations use 'e' for eye; 'br' for brow; 'm' for mouth. A superscript denotation uses 'L' for left-side; 'R' for right-side; 'T' for top; 'B' for the bottom. A subscript enumerates feature-points for the same organ.

A. Notations

Line-segments are denoted by two end feature-points or their intuitive description. For example, eye-width is denoted as *EW* or $e_1^L e_4^L$. Lip-width is denoted by *LW* or $m^T m^B$. Given a line-segment *LS*, magnitudes of the x-axis, y-axis and z-axis component are denoted respectively by $|LS|_X$, $|LS|_Y$, and $|LS|_Z$. In this paper, parameterization is illustrated using left-side of a face. The technique applies also to the right-side of the face.

B. Facial Symmetry

Facial features have an anatomical symmetry at the muscle level around the vertical axis. This symmetry causes similar changes on both sides of a face for most facial-expressions.

C. Occlusion and Head Movement

In a real-world situation, the head rotations are observed every 5 - 7 degrees [35]. In our experiment, internal states change every 15° to reduce computational overhead. This choice slightly degrades (by 1-3%) the prediction accuracy for a tradeoff of reducing computational overhead. The angle of rotation maps to one of the internal states based upon an identifiable resolution in the feature-points. Distances between the symmetry-axis and the feature-points on the nonoccluded side are used to estimate the coordinates of occluded feature-points using facial-symmetry.

In our statistical reporting of data, five occlusion states are used: 1) frontal face with no occlusion ($|\theta| < \epsilon$); 2) partial left-side or right-side occlusion (ϵ < rotation < 45°); 3) full left-side or right-side occlusion (> 45°). Internal states map to one of the five states based upon interval inclusion.

IV. Proposed Geometric Model

Facial expression analysis requires: 1) removal of the distortions caused by camera zooming; 2) removal of the distortions in the line-segments caused by head-rotations, and 3) correspondences of parameters to the changes in FAUs.

The identification of parameters invariant to head-rotations requires the use of *fixed feature-points* that act as a reference to measure the changes in orientation and lengths of the line-segments with varying facial expressions.

The motions of *active-points* that contribute to the facial expressions are: 1) vertical and horizontal motion of t b_1^L, b_2^L, b_2^L on an eyebrow; 2) vertical motions of $\{e_2^L, e_3^L\}$ in the center of an eyelid, 3) vertical and horizontal motions of m^L (lip-endpoints), and 4) vertical motions of m^T, m^B and $\{m_1^L, m_2^L\}$ (lip-midpoints). Figure 2 shows left side of the face with the required feature-points and line-segments used in the facial expression classification.

The line-segments for the facial expression analysis are: $n^B b_1^L$, $n^B b_2^L$, $n^B b_3^L$, $b_1^L b_3^L$, *EH* ($e_2^L e_3^L$: eye-height), *LH* ($m^T m^B$: lip-height), *LW* ($m_1^L m^L$: lip width), *EL* ($m^L e_c^L$: lip segment to the eye (e_c^L is the left center of eye given by $\frac{e_2^l + e_3^l}{2}$). The line-segments $n^B b_1^L$, $n^B b_2^L$, $n^B b_3^L$, $b_1^L b_3^L$ and *EL* have x-magnitudes and z-magnitudes.

The line-segments *LH* and *EH* have z-magnitudes; the line-segment *LW* has x-magnitude. With no rotation and zooming, changes in the x-magnitudes and z-magnitudes of these line-segments correspond to different facial expressions. In an actual scenario, these line-segments vary with head-rotations and image scaling due to the camera-zooming. These line-segments are mapped to parameters invariant to head-rotations and camera zooming, such that the resulting parameters vary with facial expressions only. Four line-segments, joining fixed-points, $n^B n^T$, $e_1^L e_4^L$, $n^T e_1^L$, and $n^T e_4^L$ have been used to derive parameters invariant with respect to head rotation. The effect of zooming is removed by dividing the z-magnitudes by the magnitude of the line-segment $n^B n^T$.

To minimize the effect of variation of x-coordinates during a head-rotation, the most *aligned fixed segments* are chosen that are affected similarly by the head-rotation compared to line-segments involving *active points*.

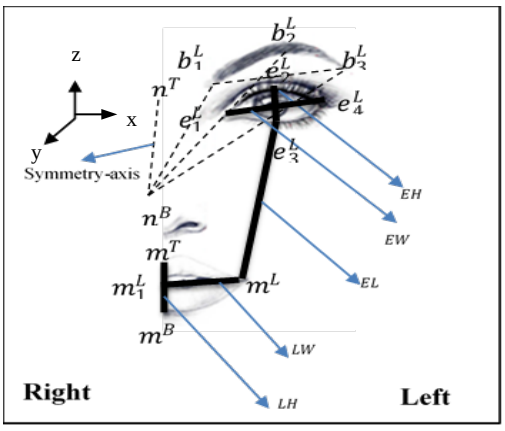

Figure 2. Facial feature-vectors

A division of line-segments by the x-magnitude of the line-segments involving the nearest fixed-points parallel to the same axis minimizes the effect of rotation and preserves the changes due to facial expressions.

The division by the segment $e_1^L e_4^L$ provides invariance for the eye-brow area. The division by x-magnitude $|\, n^T e_1^L \,|_X$ cancels the effect of head-rotation on the magnitude $|\, n^B b_I^L \,|_X$. The division by the x-magnitude $|\, n^T e_4^L \,|_X$ cancels the effect of the head-rotation on the magnitude $|LW\,|_X$.

A. Frontal Pose Estimation

The fixed feature-points nose-bottom n^B, left inner-eye e_1^L and right inner-eye e_1^R are used to establish frontal pose (see Figure 2). The ratio $|\, n^T e_1^L \,| \,/\, |\, n^T e_1^R \,| = 1$ for the frontal-pose, only altering during head-rotation. The overall estimate for the frontal pose is given by (1) where ϵ is an experimentally derived value slightly greater than zero to take care of involuntary and random head-movements.

$$1 - \epsilon \leq |\, n^B e_1^L| \,/\, |\, n^B e_1^R| \leq 1 + \epsilon \quad (1)$$

Estimation of rotation angles is based on missing landmarks on the rotated side of the face. The landmarks n^t and n^b become invisible in the complete occlusion and are visible between partial and complete occlusion. For rotation to the left or right, the ratio changes beyond $1 \mp \epsilon$.

Variations in the line-segment *LH* reflect tightening or opening of lips and mouth, and jaw-drop. It is associated with FAU 8 (lips towards each-other), FAU 10 (upper lip-raiser), FAU 16 (lower lip-depressor), FAU 17 (chin-raiser), FAU 23 (lip-tightener), FAU 26 (jaw-drop) and FAU 27 (mouth-stretcher). Variations in the line-segment *LW* reflect compression and stretching of a mouth. It corresponds to FAUs 6, 12, 14, 20, 23 and 27. These FAUs are involved in *happiness* (lip-corner and cheek-stretching obliquely up), and *sadness* (lip-corner stretching oblique downwards). Variations in the z-component $|EL|_Z$ (eye-to-lip vertical component) measure compression and stretching of cheek muscles. The decrease in $|EL|_Z$ corresponds to FAU 6 (cheek-raiser) associated with *happiness*. The increase in $|EL|_Z$ corresponds to FAU 15 (lip-corner depression) associated with negative emotions *fear, disgust* and *sadness*. The change in the magnitude of the line-segments *EW* (eye-width) and *EH* (eye-height) correspond to FAU 7 associated with *anger*. The magnitude $|EH|$ increases during *anger* due to the raising of the upper eyelid and middle eye-brow point. Variations in eye-brow length $|\, b_1^L \; b_3^L \,|$ (brow compression and stretching) correspond to FAU 1 (inner brow raiser), FAU *2* (upper brow raiser) or 4 (brow lowerer). However, only the x-component $|\, b_1^L \; b_3^L \,|_X$ is used because vertical variations in eye-brow are processed by $|n^B b_1^L|_Z$, $|n^B b_2^L|_Z$ and $|\, n^B b_3^L \,|_Z$. The increase in $|\, b_1^L \; b_3^L \,|_X$ corresponds to FAU 4 (brow-lowerer) associated with negative emotions: *fear, disgust, anger*, and *sadness.* The z-component $|n^B \, b_1^L \,|_Z$ corresponds to inner-eyebrow raising or lowering. The increase in magnitude $|n^B \, b_1^L \,|_Z$ corresponds to FAU 1 associated with *surprise*. The decrease in $|n^B b_{i(1\leq i\leq 3)}^L|_Z$ corresponds to FAUs 4 and 9 associated with negative emotions: *fear, disgust, sadness,* and *anger*. The increase in the magnitude $|n^B \, b_3^L \,|_Z$ corresponds to FAU 2 associated with *fear*. Overall, these line-segments cover FAUs 1, 2, 4, 5, 6, 7, 8, 9, 10, 12, 14, 15, 16, 17, 20, 23, 26 and 27 involved in six basic facial expressions. The overall correspondence is summarized in Table I.

TABLE I. LINE-SEGMENTS

Line-ratio	Norm. ratio	Description
R^{LH}	$\|LH\| \,/\, \|n^B n^T\|$	lip height ratio
R^{LW}	$\|LW\|_X \,/\, \|EW\|$	lip-width ratio
R^{EL}	$\|EL\|_Z \,/\, \|n^B n^T\|$	eye-to-lip ratio
R^{BW}	$\|\, b_1^L \; b_3^L \,|_X \,/\, EW$	brow-width ratio
R^{IBH}	$\|\, n^B b_1^L \,|_Z / \|n^B n^T\|$	inner brow-height ratio
R^{MBH}	$\|\, n^B b_2^L \,|_Z \,/\, \|n^B n^T\|$	mid-brow height ratio
R^{OBH}	$\|\, n^B b_3^L \,|_Z \,/\, \|n^B n^T\|$	outer-brow height ratio
R^{EH}	$\|EH\| \,/\, \|n^B n^T\|$	eye-height ratio

B. Normalized Ratios

In the beginning, the frontal pose is recorded to derive the original coordinates of feature-points and the original length and orientation of line-segments. The zooming distortion and head-rotation distortions in the x-direction are removed from the feature-points and the corresponding line-segments.

Vertical segments $|n^B b_1^L \,|_Z$, $|n^B b_2^L \,|_Z$ $|n^B b_3^L \,|_Z$, *EH* and $|EL|_Z$ are divided by $|n^B n^T|$ to derive the corresponding normalized ratios. Horizontal line-segment $|LW|_X$ and $|\, b_1^L \; b_3^L \,|_X$ are divided by $|n^T \, e_1^L \,|$ and *EW,* respectively. The normalized ratios are summarized in Table II.

TABLE II. LINE-SEGMENTS AND FAU CORRESPONDENCE

Line-seg.	FAUs	Basic emotions
LH	8, 10, 16, 17, 23, 26, 27	anger, disgust, fear, sadness, surprise
LW	6, 12, 15, 16, 20, 23	happiness and sadness
EL	6, 15	disgust, fear, happiness, sadness
EH	5, 7	anger
$\|\, b_1^L \; b_3^L \,|_X$	4	anger, disgust, fear, sadness
$\|n^B b_1^L \,|_Z$	1, 4, 9	anger, disgust, fear, sadness, surprise
$\|n^B b_2^L \,|_Z$	4, 5	fear and surprise
$\|n^B b_3^L \,|_Z$	2	fear
$n^B n^T$	Used for vertical normalizations	
$EW, \|n^T n^B\|$	Invariant with head-rotation	

C. FAU Correspondence

Table III describes conditions by combining the normalized ratios across the same or different video-frames

that are sampled periodically because facial expressions alter after few seconds. All the FAUs involved in basic facial expressions are derived using these conditions.

TABLE III. FAUs AND NORMALIZED RATIO CONDITIONS

FAUs	Condition ($n = m + k$ and $k > 0$)
#1	$R_n^{IBR} < R_m^{IBR}$
# 2	$R_n^{OBR} > R_m^{OBR}$
#4	$R_n^{IBR} < R_m^{IBR} \wedge R_n^{MBR} < R_m^{MBR} \wedge R_n^{OBR} < R_m^{OBR}$
#5, 27	$R_n^{EH} > R_m^{EH}$
#6, 12	$R_n^{LH} < R_m^{LH} \wedge R_n^{EL} < R_m^{EL}$
#7, 41	$R_n^{EH} < R_m^{EH}$
#8	$R_n^{LH} < R_m^{LH}$
#10	$R_n^{LH} > R_m^{LH}$
#15	$R_n^{EL} > R_m^{EL} \wedge R_n^{EW} > R_m^{EW}$
#16	$R_n^{LH} < R_m^{LH} \wedge R_n^{EL} > R_m^{EL}$
#17	$R_n^{EL} < R_m^{EL}$
#20	$R_n^{LW} < R_m^{LW}$
#23	$R_n^{LW} > R_m^{LW}$
#26	$R_n^{EL} > R_m^{EL}$

The increase in the ratio R^{LH} corresponds to FAU 10 (upper lip raiser), FAU 26 (jaw-drop), and FAU 27 (mouth-stretch). The decrease in the ratio R^{LH} corresponds to FAU 8 (lips towards each other), FAU 16 (lower lip-depressor), FAU 17 (chin-raiser), and FAU 23 (lip-tightener).

The increase in the ratio R^{LW} corresponds to FAU 6 (cheek-raiser), FAU 12 (lip-corner puller), FAU 15 (lip-corner depressor), FAU 16 (lower lip-depressor), and FAU 20 (lip-stretcher). The decrease in the ratio R^{LW} corresponds to FAU 23 (lip-tightener). The increase in the ratio R^{EL} corresponds to FAU 15 (lip-corner depressor); the decrease in the ratio R^{EL} corresponds to FAU 6 (cheek-raiser).

The increase in the ratio R^{EH} corresponds to FAU 5 (upper lid raiser); the decrease in the ratio R^{EH} corresponds to the FAU 7 (lid tightener) or FAU 41 (lip-stoop). The increase in the ratio R^{BW} corresponds to FAU 4 (brow-lowerer). The increase in the ratio R^{IBR} corresponds to FAU 1 (inner eye-brow raiser); the decrease in the ratio R^{IBR} corresponds to FAU 4 (brow-lowerer). The increase in the ratio R^{OBR} corresponds to FAU 2 (outer eye-brow raiser); the decrease in R^{OBR} corresponds to FAU 4 (eye-brow lowerer).

A simultaneous decrease in the ratio R^{LH} and an increase in the ratio R^{EL} correspond to the activation of FAU 16 (lower-lip depressor). Simultaneous decreases in the ratios R^{LH} and R^{EL} correspond to the activations of FAU 12 (lip-corner puller) and FAU 6 (cheek-raiser). Simultaneous increases in the ratios R^{EL} and R^{EW} correspond to FAU 15 (lip-corner depression). Simultaneous decreases in the ratios R^{IBR}, R^{MBR} and R^{OBR} and, increase in the ratio R^{BW} correspond to the activation of FAU 4 (eye-brow lowerer).

V. IMPLEMENTATION AND EXPERIMENTATION

RaFD database [20] was used for training and comparison of results between geometric modeling and CNN-based model. For the online video capturing, three frames per second were used for the facial expression analysis. Epochs of 200 frames were used because the experimental data show that the accuracy of the facial expression recognition stabilizes around 200 frames.

A. CNN Architecture

The implemented CNN-based model is a cascade of three hidden layers: conv-32, conv-64 and conv-128, followed by a Softmax layer. Each *conv-m* layer contains *m* filters to extract different orientations. The conv-128 layer provides a sub-classification of textures. After each convolution layer, there is a max-pooling layer for the subsampling of images. Each max-pool layer has a 2 × 2 pixel window.

After applying the Locality-Sensitive Hashing (LSH) [12] and Gabor filter [13], the processed images are passed to the network of convolution layers through the input layer. LSH is a dimension reduction technique that maps the pixels with similar values in the same bucket. Gabor filter preserves the texture directionality. The hidden layers extract facial features and reduce the dimensions. The fully connected layer combines the matrix-derived after the last hidden layer into one vector, and the Softmax layer extracts the output from the vector.

Each cropped image is scaled to 56 × 56 pixels. The data-size after the conv-32 layer is 56 × 56 × 32 pixels, and the output of first max-pooling layer after the conv-32 layer is 28 × 28 × 32 pixels. The output of the second max-pooling layer is 28 × 28 × 64 pixels. The output of the last hidden layer is 14 ×14 ×128. The output of the following max pooling layer is 7 × 7 × 128 pixels. Extracted features are concatenated by adding a fully connected layer at the end.

B. Database and Video Processing

RaFD dataset was used for measuring the performance of the CNN-based model for various static alignments in different poses [20], [36]. Compared to other curated facial expression databases u [17]-[19], RaFD gives comprehensive facial-expressions for 67 models (for both genders) with multiple camera angles and adjustment of lighting conditions.

CNN model was also executed in wild for the frontal pose and compared against the results of RaFD dataset to derive the comparative deterioration of the *recall* as defined in (2).

$$Recall = \frac{true\ positive}{true\ positive\ +\ false\ negative} \quad (2)$$

The hybrid model was executed in the wild. The results are summarized in Tables IV, V, and VI, respectively. Tables IV and VI show the recall values of CNN-model with RaFD dataset and the proposed hybrid model in wild, respectively. Table V shows the confusion matrix for CNN model for frontal pose in the wild.

C. Performance Evaluation and Discussion

Table IV illustrates CNN based prediction, even for a cured RaFD database, deteriorates quickly due to the unavailability of discriminatory feature-points on the

occluded part of the face. The deterioration varies from 48% for sadness to 41% for happiness for complete occlusion.

TABLE IV. RECALL IN CNN MODEL WITH RADB DATASET

	Right complete occlusion	Right part occl.	Front no occl.	Left part occl.	Left complete occlusion
sadness	49%	83%	97%	79%	48%
disgust	54%	81%	98%	88%	63%
anger	53%	81%	96%	87%	64%
fear	51%	86%	95%	81%	55%
surprise	57%	84%	98%	90%	53%
happiness	59%	85%	99%	92%	62%
neutral	54%	82%	95%	79%	51%

TABLE V. CONFUSION MATRIX - CNN MODEL (FRONTAL POSE) IN WILD

	sad. %	disg. %	ang. %	fear %	sur. %	happ. %	neu.
sadness	74.5	0.1	8.0	12.3	0.9	0.7	3.5
disgust	0.7	92.4	1.4	1.1	1.3	1.7	1.4
anger	6.4	2.3	79.3	2.5	1.6	2.4	5.5
fear	7.2	0.6	6.1	82.3	1.2	0.8	1.8
surprise	1.8	0.7	2.6	5.2	86.9	1.7	1.1
happines	1.4	0.2	2.2	2.5	3.0	87.2	2.5
neutral	10.2	0.2	4.2	5.7	2.2	3.7	73.8

TABLE VI. RECALL IN HYBRID MODEL IN WILD

	Right complete occlusion	Right part occl.	Front no occl.	Left part occl.	Left complete occlusion
sadness	57%	68%	75%	69%	59%
disgust	70%	81%	92%	82%	70%
anger	73%	75%	79%	77%	76%
fear	66%	75%	82%	76%	67%
surprise	71%	74%	87%	76%	75%
happines	75%	79%	87%	81%	77%

Comparison of Table IV and Table V illustrates that the accuracy of facial expression classification deteriorates in the wild even for the frontal pose: more for sadness (around 22%) and the least for disgust (around 6%). Even neutral face is labeled as sad for 10% of the time in the wild. The reasons for this deterioration are: 1) mixing of facial muscles and feature-points for negative facial expressions, *sadness*, *fear* and *anger*, in real-time expressions; 2) variations in the intensity level of the expressed facial expressions in real-time; 3) continuous random head-motions during real-time facial-expressions causing noise; 4) uneven ambient lighting conditions with shadows obscuring feature-points; 5) randomly picking the video-frame may not correspond to the apex image corresponding to a facial-expression [30].

The facial expressions for the negative emotions: *sadness*, *fear*, and *anger* are often confused due to 1) the presence of common facial muscles; 2) the mixing of facial expressions in real-time; 3) improper temporal labeling during transition of a negative facial expression to another; 4) uncontrolled thought patterns affecting involuntary facial expressions in real-time. Another problem is that CNN is trained using fixed alignments, and a head-movement is approximated to one of the fixed poses.

Comparison of the occluded parts in Table IV and Table VI shows that the hybrid model outperforms CNN-based prediction even for the curated RaFD dataset for beyond the partial occlusion. The improvement is 8% for sadness (minimum) to 21% for the happiness (maximum). In a multi-party interaction, where the change in the line-of-view may cause extreme occlusion, the hybrid model provides better accuracy and information.

The current scheme can be further improved by smoothening the derived facial-expression sequence and predicting the next facial-expression using Dynamic Bayesian Network (DBN), the knowledge of average duration of facial-expressions during emotional conversation, and sampling more video-frames for near-apex facial expressions.

VI. CONCLUSION AND FUTURE WORK

Head-motions during conversational gestures and multi-agent interactions cause extreme occlusion of one side of facial features. Automated feature-extracting and deep learning schemes are limited by the facial feature detections. Their performance degrades during extreme occlusion due to the nonavailability of discriminatory feature-points. Facial symmetry reconstructs the occluded discriminatory feature points. Combining CNN based schemes with the proposed geometric modeling improves the performance in such a scenario by 8% – 21% beyond the partially occluded state.

We are currently investigating the DBN on a sequence of facial-expressions to smoothen out the errors due to image frames missing the apex image for the corresponding facial expressions [30].

REFERENCES

[1] M. I. Yenilmez, “Economic and social consequences of population aging the dilemmas and opportunities in the twenty-first century,” Applied Research in Quality of Life, vol. 10, no. 4, pp. 735-752, Dec. 2015, doi: 10.1007/s11482-014-9334-2.

[2] D. H. García, P. G. Esteban, H. R. Lee, M. Romeo, E. Senft, and E. Billing, “Social Robots in Therapy and Care,” 14th ACM/IEEE International Conference on Human-Robot Interaction, Daegu, South Korea, 2019, pp. 669-670.

[3] C. Peter and R. Beale (eds), “Affect and Emotion in Human-Computer Interaction: From Theory to Applications,” LNCS 4868, Berlin / Heidelberg: Springer-Verlag, 2008.

[4] M. Ghayoumi, M. Thafar, and A. K. Bansal, “A Formal Approach for Multimodal Integration to Derive Emotions,” Journal of Visual Languages and Sentient Systems, vol. 2, pp. 48-54, Oct. 2016, doi: 10.18293/DMS2016-030.

[5] F. Rothganger, “Computation in the Wild,” https://www.osti.gov/servlets/purl/1644432, [accessed date; June 12, 2021.

[6] Y. Zong, W. Zeng, X. Huang, K. Yan, J. Yan, and T. Zhang, “Emotion recognition in the wild via sparse transductive transfer linear discriminant analysis, Journal of Multimodal Interfaces, vol. 10, 2016, pp. 163-172. doi:10.1007/s12193-015-0210-7.

 ISBN: 978-1-61208-882-2

[7] C. M. Lee and S. Narayanan, “Towards Detecting Emotions in Spoken Dialogs,” IEEE Transactions on Speech and Audio Processing, vol. 13, no.2, pp. 293-303, March 2005.

[8] K. Han, D. Yu, and I. Tashev, “Speech Emotion Recognition Using Deep Neural Network and Extreme Learning Machine,” 15th Annual Conference of the International Speech Communication Association, Singapore, Sept. 2014, pp. 223-227.

[9] A. Singh and A. K. Bansal, “Towards Synchronous Model of Nonemotional Conversational Gesture Generation in Humanoids, Computing Conference, London, UK, July 2021, in press.

[10] M. Fernandez-Dols, H. Wallbott, and F. Sanchez, “Emotion Category Accessibility and the Decoding of Emotion from Facial-expression and Context,” Journal of Nonverbal Behavior, vol. 15, no. 2, pp. 107-123, 1991.

[11] Y. Huang, F. Chen, S. Lv, and X. Wang, “Facial Expression Recognition: A Survey,” Symmetry, vol. 11, no. 10, Article 1189, Sept. 2019, doi: 10.3390/sym11101189.

[12] M. Ghayoumi and A. K. Bansal, “An Integrated Approach for Efficient Analysis of Facial expressions,” The 11th International Conference on Signal Processing and Multimedia Applications, Vienna, Austria, Aug. 2014, pp. 211-219.

[13] M. Ghayoumi and A. K. Bansal, “Emotions in Robot using Convolutional Neural Network,” International Conference on Social Robotics (ICSR 2016), Kansas City, KS, USA, Nov. 2016, pp. 285-295.

[14] P. Ekman and W. V. Friesen, “Nonverbal Behavior,” Communication and Social Interaction, P. F. Ostwald, (Editor), New York, NY: Grune & Stratton, pp. 37-46, 1977.

[15] R. Plutchik, “Emotion: A Psychoevolutionary Synthesis,” New York, NY: Harper & Row, 1980.

[16] K. Seshadri and M. Savvides, “Towards a Unified Framework for Pose, Expression, and Occlusion Tolerant Automatic Facial Alignment,” IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 38, no. 10, Oct. 2016, pp. 2110-2122, doi: 10.1109/TPAMI.2015.2505301.

[17] P. Lucey, J. F. Cohn, T. Kanade, J. Saragih, Z. Ambadar and I. Matthews, “The Extended Cohn-Kanade Dataset (CK+): A Complete Dataset for Action Unit and Emotion-specified Expression," International Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), San Francisco, CA USA, June 2010, pp. 94-101, doi: 10.1109/CVPRW.2010.5543262.

[18] R. Gross, I. Matthews, J. F. Cohn, T. Kanade, and S. Baker, “Multi-PIE,” In Proceedings of the Eighth IEEE International Conference on Automatic Face and Gesture Recognition, Vol. 28, No. 5, May 2008, pp. 807-813, doi: 10.1016/j.imavis.2009.08.002.

[19] M. J. Lyons, M. Kamachi, and J. Gyoba, "Japanese Female Facial-expressions (JAFFE), " 1998, doi: 10.5281/zenodo.3451524.

[20] O. Langner, R. Dotsch, G. Bijlstra, D. H. J. Wigboldus, S. T. Hawk, and A. van Knippenberg, “Presentation and validation of the Radboud Faces Database,” Cognition & Emotion, vol. 24, no. 8, pp. 1377-1388, Dec. 2010, doi: 10.1080/ 026999 30903485076.

[21] L. Zhang, B. Verma, D. Tjondronegoro, and V. Chandran, “Facial Expression Analysis Under Partial Occlusion: A Survey,” ACM Computing Surveys, vol. 51, no. 2, Article No. 25, Apr. 2018, doi: 10.1145/3158369.

[22] S. S. Liu, Y. Zhang, and K. P. Liu, “Facial expression Recognition under Random Block Occlusion Based on Maximum Likelihood Estimation Sparse Representation,” International Joint Conference on Neural Networks (IJCNN 2014), Beijing, China, July 2014, pp. 1285-1290.

[23] L. Shuaishi, Z. Yan, and L. Keping, “Facial-expression Recognition under Partial Occlusion Based on Weber Local Descriptor Histogram and Decision Fusion,” The 33rd Chinese Control Conference, Nanjing, China, July 2014, pp. 4064-4068.

[24] Q. Cheng, B. Jiang, and K. Jia, “A Deep Structure for Facial Expression Recognition under Partial Occlusion,” The Tenth International Conference on Intelligent Information Hiding and Multimedia Signal Processing (IIH-MSP 2014), Kitakyushu, Japan, Aug. 2014, pp. 211-214.

[25] J. Y. R. Corenjo and H. Pedrini, “Recognition of occluded facial expressions based on CENTRIST features,” IEEE International Conference on Acoustics, Speech and Signal Processing, 2016, pp. 1298-1302.

[26] R. Li, P. Liu, K. Jia, and Q. Wu, “Facial Expression Recognition under Partial Occlusion Based on Gabor Filter and Gray-level Co-occurrence Matrix,” The International Conference on Computational Intelligence and Communication Networks, Jabalpur, India, Dec. 2015, pp. 347-351.

[27] Y. Li, J. Zeng, S. Shan, and X. Chen, “Occlusion Aware Facial Expression Recognition Using CNN with Attention Mechanism,” IEEE Transactions on Image Processing, vol. 28, no. 5, pp. 2439-2450, May 2019.

[28] F. Zhao, J. Feng, J. Zhao, W. Yang, and S. Yan, “Robust LSTM-Autoencoders for Face De-Occlusion in the Wild,” vol. 27, no. 2, pp. 778-790, Feb. 2018.

[29] Y. Miyakoshi and S. Kato, “Facial emotion detection considering partial occlusion of face using Bayesian network,” IEEE Symposium on Computers & Informatics, Kuala Lumpur, Malaysia, March 2011, pp. 96-101.

[30] A. Cruz, B. Bhanu, and N. S. Thakoor, “Vision and Attention Theory-Based Sampling for Continuous Facial Emotion Recognition,” IEEE Transactions of Affective Computing, vol. 5, no. 4, pp. 418-431, Oct-Dec. 2014.

[31] T-H S. Li, P-H Kuo, T-N Tsai, and P-C Luan, “CNN + LSTM Based Facial Expression Analysis Model for a Humanoid Robot,” IEEE Access, vol. 7, pp. 93998-94011, July 2019, doi: 10.1109/ACCESS.2019.2928364.

[32] S. Derrode and F. Ghorbel, “Shape Analysis and Symmetry Detection in Gray-level Objects using the Analytical Fourier-Mellin Representation,” Signal Processing, vol. 84, no. 1, pp. 25-39, Jan. 2004.

[33] S. Kondra, A. Petrosino, and S. Iodice, “Multi-scale Kernel Operators for Reflection and Rotation Symmetry: Further Achievements,” IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW 2013), Portland, OR, USA, June 2013, pp. 217-222, doi: 10.1109/CVPRW. 2013.39.

[34] M. Ghayoumi and A. K. Bansal, “Real Emotion Recognition by Detecting Symmetry Patterns with Dihedral Group,” Third International Conference on Mathematics and Computers in Sciences and in Industry (MCSI 2016), Chania, Greece, Aug. 2016, pp. 178-184, doi: 10.1109/MCSI.2016.041.

[35] M. Amiri, G. Jull, and J. Bullock-Saxton, “Measuring range of active cervical rotation in a position of full head flexion using the 3D Fastrack measurement system: an intra-tester reliability study,” Manual Therapy, vol. 8, no. 3, pp. 176-179, Aug. 2003.

[36] A. R. Dores, F. Barbosa, C. Queirós, I. P. Carvalho, and M. D. Griffiths, “Recognizing Emotions Through Facial Expressions- A Large Scale Experimental Study,” International Journal of Environmental Research and Public Health, vol. 17, Article 7420, doi:10.3390/ijerph17207420

 ISBN: 978-1-61208-882-2

Vision-based Inspection System for Ornamental Stone Using a Weighted Hybrid Ensemble Classifier

Liliana Antão*, Rui Pinto*, Manuel J. Ferreira†, Tiago Pinto†, Gil Gonçalves*
*Research Center for Systems and Technologies
Faculty of Engineering, University of Porto
Rua Dr. Roberto Frias, 4200-465 Porto, Portugal
Email: { lpsantao, rpinto, gil }@fe.up.pt
†Neadvance – Machine Vision
Rua do Parque Poente, Lote 35 4705-002 Sequeira, Braga, Portugal
Email: { mferreira, tpinto }@neadvance.com

***Abstract*—With the increasing demands of customers in the ornamental stone industry, both in terms of the individual specifications of each product and in the delivery times, it is necessary to constantly adapt the manufacturing processes and their inherent complexity and, consequently, the automated systems that are essential to them. There is a strong movement of research in areas capable of generating non-destructive testing techniques applied to production systems in this sector. Currently, one of the main problems occurs during the ornamental stone slab polishing phase, where there is the need to monitor the polishing quality and diagnose possible defects in the surface of the slab. This can be used as feedback for self-correction and optimization of variables and process parameters in the polishing equipment. In this paper is proposed a monitoring system, based on machine vision techniques, used to detect defects in the surface of polished ornamental stone slabs. This approach is based on a weighted hybrid ensemble classifier, relying on image processing techniques and a Convolutional Neural Network. Results show that the ensemble classifier outperforms related classifiers.**

***Index Terms*—Ornamental Stone, Ensemble Classifier, Convolutional Neural Network, Machine Vision.**

I. Introduction

Regarding the natural and ornamental stone industrial sector, there is a positive response of ornamental stone manufacturing companies, especially in Portugal, in incorporating Industry 4.0 related practices and technologies in their production [1]. This allows them to enhance and achieve added value in this significant industrial sector since the transformation of rock minerals has enormous importance in the Portuguese market. In the stone extraction and transformation industry, typically, the first step marks the extraction of large blocks from quarries. There are several types of ornamental stone, such as marble, granite, limestone, among others. Then, the cutting and sawing process transforms these blocks into slabs. These slabs go through a polishing process to remove the cutting process's imperfections and restore brightness to the slab. Later, the slabs can suffer a final cut for smaller slabs or tiles, which will later be used in pieces, such as countertops and kitchen or bathroom tops. Finally, the process culminates in the packaging of the final product.

Ornamental stone manufactured goods, such as countertops and kitchen or bathroom tops, must satisfy specific aesthetic requirements, namely stone surface status or its reflecting properties, which can be accomplished by the polishing process. According to Bonifazi & Marinelli [2], there is not a defined way to know what constitutes a good polishing process. This greatly depends on the type of stone, its mineral composition, and textural attributes. Such judgment relies on human expertise. It can be influenced by several characteristics, such as mineral grains size and their relative arrangement, background color, presence of veins and plagues, cultural level of the human inspectors, and final destination of the stone manufactured good. Fig. 1 represents examples of ornamental stone manufactured goods.

Fig. 1. Ornamental stone manufactured goods, such as kitchen/bathroom tops.

Mathielo & Bolonini [3] conducted a recent study in Brazil, where they concluded that 77% of companies in the ornamental stone sector perform analysis of the quality of the polished surfaces of stone via visual inspection, only 3% do this by measuring the brightness surface and 20% use both. This demonstrates the subjective character and the possible incompatibility between the analyzes of different human inspectors regarding the polishing quality of the finished surface. In this sense, it is necessary to adapt the stone manufacturing processes and their inherent complexity. Consequently, one needs to adapt the automated systems that are essential to them, namely automated methods to evaluate the efficiency of the polishing process in ornamental stone. Monitoring using sensors is a common method of recognition and adaptation to the environment by automatic systems, and it can be assumed

 ISBN: 978-1-61208-882-2

that computer vision is one of the most versatile technologies for this purpose [4].

Computer vision is based on the interpretation of images captured by non-invasive optical sensors, in order to extract useful information from a real scenario for analysis and/or process control. This technology allows identification and analysis of geometries, position detection, supervision, quality control, and measurement (speed, deformations, temperature, etc.). As a non-contact measuring system, it does not suffer considerable wear and can operate continuously even in hostile environments. For all these reasons, it is understandable that such a solution is of interest in a wide range of applications, such as closed-loop control processes [5].

In addition to computer vision, there is a research hype in areas capable of generating other types of Non-Destructive Testing (NDT) techniques for monitoring purposes, applied to the several stages in the ornamental stone production process. Montiel-Zafra *et al.* [6] present an impact-echo method to analyze the internal quality of ornamental stone blocks right after being extracted from the quarry.On the other hand, already on the polishing stage, Maria *et al.* [7] propose a wavelet technique to detect defects using images of ornamental limestone slabs.

Considering computer vision-related approaches for monitoring and defect detection, most of the literature work focuses on exploring Machine Learning (ML) classification techniques. This work intends to address the application of computer vision with ML for monitoring the quality of a stone slab polishing process by detecting defects on the stone's surface. The polishing defects may be seen as the presence of scratches and irregularities in the stone surface, which could be introduced by the polishing process itself or were already present in the stone, and the polishing process could not remove them. In this sense, we propose an approach, based on computational vision, to monitor complex defects and errors of high detection complexity during the ornamental limestone polishing process. This approach allows the verification and analysis of defects in polished ornamental limestone in a non-destructive way. This classification can later be used as feedback for self-correction for the regulation/optimization of the polishing variables and process parameters.

This paper is organized into four more sections. Section II provides the state of art about machine vision architectures and techniques for defect inspection in ornamental stone. Section III provides a detailed characterization of the proposed approach, a weighted hybrid ensemble classifier based on image processing and a Deep Learning model. Section IV discusses the experimental results achieved. Finally, Section V concludes the paper, stating final remarks about the work presented and provides orientations for future work.

II. Related Work

Tantussi & Lanzetta [8] provided a review about optical methods for stone surface inspection. Authors mention four main non-contact surface inspection methods, namely optical profilometry, glossmetry, riflectometry, and artificial vision. This section focuses on the literature review regarding artificial vision-based approaches for automated inspection of ornamental stone surface for defects.

It was mentioned before the difficulty to assess the quality of a stone surface due to the subjective nature of this classification, mostly by considering visible pictorial attributes and overall aesthetic features. Efforts have been made to propose methodologies and techniques able to quantify the aesthetic quality level of stone-based manufactured goods [9]. In this approach, Bonifazi *et al.* tried to assess the quality of the stone by evaluating the degree of polishing of the stone surfaces and the presence of defects, using image processing techniques. They also classify the main defects being: 1) Grooves; 2) Fissures and/or holes, and 3) Mineral inclusions. On the other hand, Yarahmadi *et al.* [10] present a new approach for quantifying the quality of stone products in quarries and processing plants.

In 2005, Lee *et al.* [11] propose an automated process for inspection of polished stone in order to detect process-induced defects, i.e., tooling marks induced by the polishing machine. The detected defects are characterized and used for adaptive control of the polishing process. The authors used the classical Circle Hough Transform (CHT) algorithm after collecting images of the surface of the stone since polishing tooling marks and scratches are distinguishable from natural flaws, characterized by their circular geometric form. So, this becomes a circle detection type of problem. The algorithm was tested using a range of images presenting defects encountered in the inspection of polished stone surfaces, obtaining good results. Issues presented were the optimization of the probability estimation, accuracy, and the relationship between speed and the proportion of edge pixels belonging to circular features to the total number present.

In 2012, Bianconi *et al.* [12] proposed an expert system for automatic classification of granite tiles through computer vision. The authors experimented with several classifiers (supervised ML) using a dataset of images. Classification takes into consideration both color and texture. Results show that the methods considered provide high granite classification accuracy, while Support Vector Machines (SVM) outperforms other methods. In 2013, Martínez *et al.* [13] proposed an automated classification approach, using ML techniques based on numeric variables obtained from 2D and 3D images captured by a linear 2D camera and a 3D laser scanner. Authors implemented both supervised and unsupervised ML techniques, such as SVM and Multilayer Perceptron neural networks (MLP), and cluster analysis and self-organizing maps. Results show that the error of automated classification was lower than for manual classification.

More recently, in 2018, Iglesias, Martínez & Taboada [14] proposed an automated inspection system for examining slate slabs based on capturing data with a 3D color camera and studying slate slab traits using computer vision algorithms. The authors tested the method on real slate slabs, which were previously classified by a human expert. Results show that the laboratory prototype system performed well, as the inspection

 ISBN: 978-1-61208-882-2

algorithms were able to accurately detect the same traits as the human expert, except for surface irregularities. Also, Ramil *et al.* [15] proposed a back-propagation Artificial Neural Network (ANN) in order to obtain the rapid and reliable identification of forming minerals in granitic rocks by means of RGB images. The results obtained, though preliminary, led to a high degree of correct identification of the forming minerals for three different granitic types.

III. Proposed Approach

To obtain a vision-based system that is able to correctly classify ornamental stone slabs as defective or not, we propose an approach based on a weighted hybrid ensemble classifier. Classification of defective slabs corresponds to the detection of defects on the surface of the slab. The overview of this approach is presented in Fig. 2. The ensemble is composed of two different classifiers, one based on traditional image processing techniques, such as adaptive filters applied according to structural and statistical methods, and another one based on a Deep Learning model, namely using a Convolutional Neural Network (CNN).

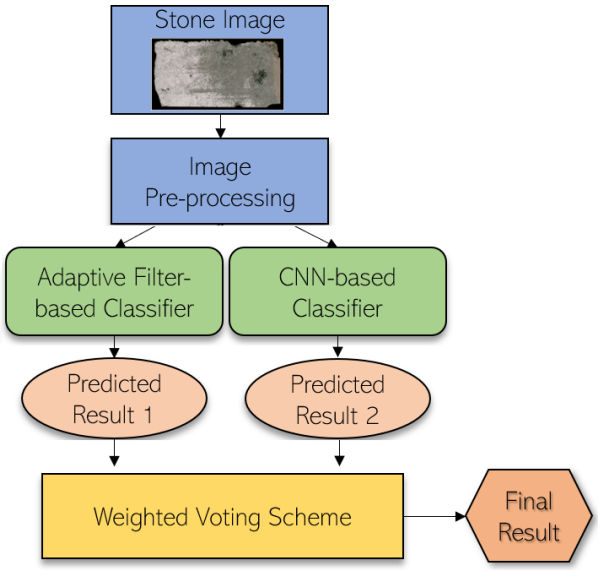

Fig. 2. Overview of the proposed hybrid classifier ensemble approach.

The inspection system starts by receiving an image of the ornamental stone's slab. This image is pre-processed in order to retrieve the biggest region of interest and feeding it as input to the classifiers. Then, each classifier obtains its predicted result separately for the input image. These predicted results are then summarized by applying a weighted voting scheme, obtaining the final result for that specific stone's surface. In this case, the result can be *OK* (no or non-significant defects) or *NOK* (significantly defective). Following Bonifazi *et al.* [9] guidelines, defect detection includes holes, fissures, decays, grooves, or deviated mineral inclusions. These are significant when occupying more than 0.05% of the region of interest of the stone surface.

A. Image Pre-processing

The first step to allow the system to correctly inspect the stone slab is provided by an image pre-processing module by converting the raw image of the stone slab into the desired input to the classifiers. Training/testing a model on raw images usually leads to bad classification performances [16], so this step helps to improve the classifier performance. Also, pre-processing techniques allow a faster training process. To allow the classification models to consider only the stone's surface, transitions between the slab and the image background or other elements present in the image are removed. Not only are those transitions not needed for this problem, but they may also cause a misdirection in the classification models since they introduce certain non-desired patterns, impacting the performance of the classifiers.

For this, it is performed segmentation of the tile to separate the stone slab surface from the image background. First, the original image is converted to grayscale, and then Gaussian filters are applied to eliminate noise while blurring the patterns and details inside the stone. Afterward, a linear threshold is applied, finally detecting the points that represent the outlines of the stone slab, using the same method proposed by [17]. To optimize memory usage, a simple approximation method can be used, which removes all redundant points and compresses the slab contours. With all the contours found, their features are evaluated, and the general external outlines of the slab stone are obtained. This step is important as one stone may appear split in more than one slab due to top to bottom fissures, and therefore return more than one external contour. In these cases, each part of the stone is evaluated separately to perceive which part needs correction, if any. This way, the pre-processing module separates the original stone image in two, allowing the classifiers to analyze the stone's defectiveness as distinct surfaces.

By using and pairing the external contour points obtained for each stone slab, it is possible to generate all rectangles that can be built using those pair of contour points. This enables to quickly reach the quadrilateral Region Of Interest (ROI), the biggest square or rectangle inside the stone slab, without any portion of the background. The ROI is then used as input to the classifiers, allowing to provide only stone surface information as input. After the initial rectangles are generated, the next set of rectangles are created by using the second point as the reference and so on.

An example of some possible rectangles is shown in Fig. 3 c). With all the rectangles generated, they are sorted by their area in descending order, being that the first rectangle in the list is the biggest one. Besides the area, additional criteria were added as sometimes the quadrilateral regions generated still included some background. Given this, a given rectangle was only selected if across its entire area there were no black dots. If any point in the perimeter of a rectangle is on a spot in which the respective pixel is 0, the respective rectangle is not valid. The rectangle that passes the criteria and has the biggest area is the desired one.

 ISBN: 978-1-61208-882-2

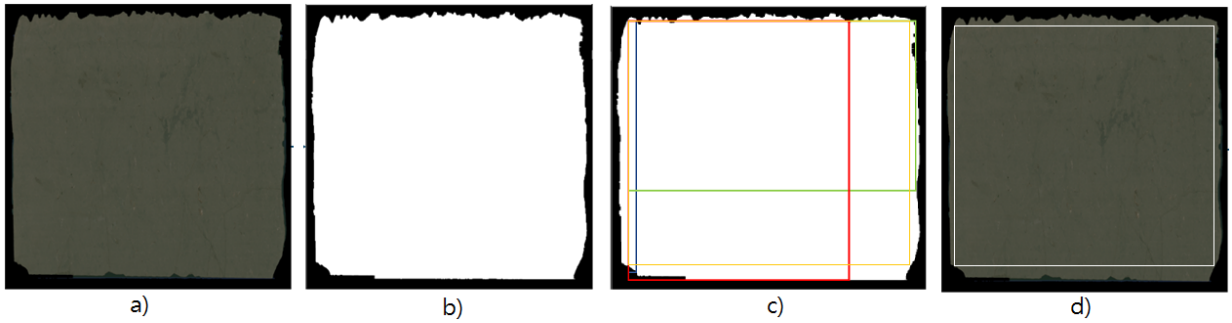

Fig. 3. Pre-processing steps: a) Original stone image, b) Image after segmentation, c) All possible rectangles; and d) Final rectangle chosen.

All the pre-processing steps performed and explained above can be seen in Fig. 3. The white square represented in Fig. 3 d) corresponds, approximately, to the biggest rectangle regarding the area, in the exemplified stone slab. The ROI is the region inside the limitation provided by the rectangle, used next as input for the classification models.

B. Weighted Hybrid Ensemble Classifier

To develop a system for inspection and detection of imperfections on the surface of the ornamental stone slabs, after obtaining the image in the desired format, the ensemble model is considered. A hybrid ensemble model is used to improve the system's performance and robustness. Compared to classifiers working individually, classifiers working together usually have the potential for a better performance.In this specific scenario, given the diversity in types of stones, this is particularly advantageous, as the limitations of one classification algorithm to a type of stone can be compensated by other classifiers.

The proposed ensemble model is composed of three main components: 1) Adaptive Filtering-based classifier that uses traditional image processing techniques with statistical and structural adaptations; 2) a Deep Learning model, particularly a Convolutional neural network (CNN-based classifier); 3) a decision-making scheme based on weighted voting. These three components are detailed in the following sections.

1) Adaptive Filtering-based classifier: Traditional filtering methods are utilized for a low cost, low level, and lightweight approach for inspection systems, as they enable to detect distinct features with no need for labeled data. Overall, these methods usually require a reference pattern without deformities, i.e., a threshold value that delimits the normal surface from the defective one. Typical filters are more suitable for images that are patterned and have periodic properties.

Considering the ornamental stone industry, where different types of stone can be extracted and used, applying a simple filtering method with static thresholds would result in poor classification performances. For instance, with marble slabs, the color and texture are typically light and homogeneous throughout its usable area, while for slate, although smooth, its texture does not follow a pattern, and its color is usually very dark. Granite, on the other hand, has a pattern of minerals along its surface, being highly variable in color and pattern (typically heterogeneous). Also, even considering the same type of stone, there may be found clear differences, according to the quarry. This difference in color and uniformity levels means that a given threshold defined as the best for one specific type of stone is not directly applicable to another. Thus, filtering methods alone are not suitable, and normally structural techniques are considered. By combining different algorithms, the defective regions can be distinguished from the stone's natural surface.

Given this, for the proposed classifier, the first step is to differentiate the basic pattern of each stone's ROI from possible defects. This step is important, as several images have innate patterns, not derived from anomalies, which could be wrongly perceived as a defect since they deviate from the normal pattern of the stone. For this a thresholding technique is applied to the image of the stone previously transformed into grayscale, using an adaptation of the Otsu method, called the valley-emphasis method [18]. In this method, the optimal threshold value is selected automatically using the images' gray-level histogram by applying the methods described in [18]. Therefore, the threshold is adapted according to the anomalies' characteristics to be isolated from each image.

After thresholding, the Sobel filter was used, followed by an oriented non-maximal suppression for edge detection. This is important to detect boundaries between the base pattern of the stone and anomalies. Then, morphological filters are applied to better identify and quantify the presence of defects on the stone surface. Each defect is identified, and its entity quantified in defect area and perimeter and the ratio of the surface area of the defect in respect to the ROI. This quantification allows checking if the detected anomaly complies with the established criteria mentioned before in Section III. If one or more defects detected pass these criteria, then the stone slab is classified as *NOK* (i.e., defective). In Fig. 4, some results from this classifier are presented. All these algorithms that comprise the adaptive filtering-based classifier are implemented in Python and OpenCV functions.

2) CNN-based classifier: As several images of stone slabs are available, a Deep Learning model is created and trained in order to classify the images as defective *NOK* or non-defective *OK*. For this, a CNN architecture is used since it is one of the best techniques for feature representation. It is

 ISBN: 978-1-61208-882-2

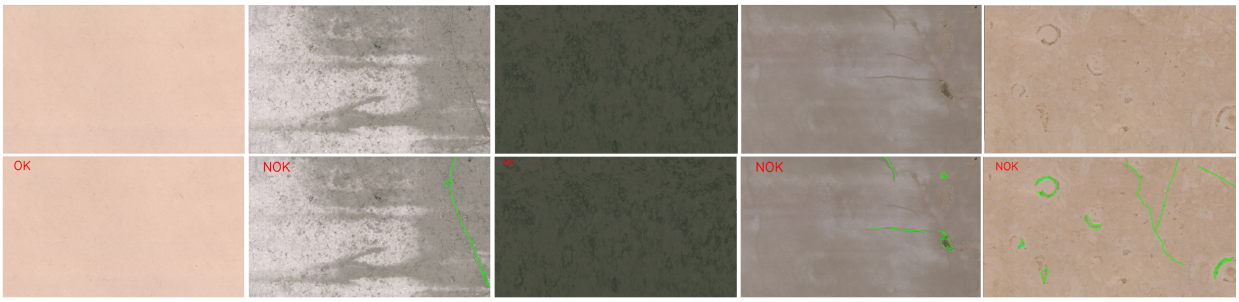

Fig. 4. Extracted ROIs of original stone images fed to the classifier (top), and corresponding classification and detection result (bottom).

easier to train than other ANN models, as they have many fewer parameters than fully connected networks with the same number of hidden units. To choose the final model, several different classifiers were considered with distinct combinations of architecture blocks and hyperparameters, randomly picked. The search space for the model was composed by four different encoders: DenseNet121, DenseNet201 [19], Inceptionv3 [20], resnet101v2 [21]. Combinations include average or max pooling, between 1 and 4 dense layers with a number of units in a Range(Start=128, End=2048, step=128). Finally, four different dropout values were tested: 0, 0.1, 0.2 or 0.3.

Given the search space defined above, each generated model with a combination of different parameters was trained three times in a dataset with a wide variety of defective and non-defective ornamental stones (more details on this dataset in Section IV). The data was divided into 70% of the images for training, 20% for validation, and the remainder used as test. The validation loss (categorial cross-entropy) was obtained from the average loss between the three runs and was used as the main metric for the model selection. The model selection was performed inside the training procedure and is divided into two stages: stage *i* that entails model elimination, where the models that performed badly were excluded; and stage *ii*, where the models resulted from the first stage are compared in performances on the test dataset.

For stage *i* of the model selection, the ten best models with the lowest average validation loss over the three trains were chosen for the final selection stage. Then, in stage *ii* of selection, those top ten models were evaluated regarding their F1-scores in the test dataset, where the final model selected had the highest score. The final model architecture achieved based on this model selection process had the following parameters: Densetnet121 as encoder, an average pooling method, and one dense layer with 1024 units and 0.1 of dropout. An Adam optimizer was used, and a learning rate of 0.0001 was selected. This final CNN was implemented in Python, using the Deep Learning framework Keras with Tensorflow backend.

3) Weighted Voting Scheme: A weighted voting ensemble is used as final decision-making for this system. Each model makes a prediction (votes) for each test instance, and the final output prediction is the one that receives more than half of the votes. In this case, only happening when both classifiers agree on the predicted result, which translates into a typical Majority voting algorithm. When the classifier models disagree, we increase the importance of one model above the other according to its performance.

This results in an approach where each model has a different significance, unlike majority voting. In this voting scheme, weights are determined according to the classification performances for each type of stone, defined by its initial characteristics, namely its heterogeneity or homogeneity. This criterion for weight setting is defined after the analysis of each classifier's performance, where it is clear that the stone surface's uniformity or lack of is the characteristic with the most impact. The scheme associates each trained classifier with a distinct weight according to its classification performance in the validation set for each type of stone. The final result for each input is done based on the highest weighted votes. If it is established that any classifier can make more confident predictions for a specific type of stone than the other, it is advisable to increase their weight to obtain more successful results.

IV. System Validation

In Portugal, the main mining district of ornamental limestones is the Maciço Calcário Estremenho (MCE). Limestones are fine to coarse-grained calciclastic sparitic rocks (rudstones and grainstones), i.e., formed by grains cemented by small amounts of translucid calcite [22]. There are several types of limestones, but the ones considered in this work are the *Cadoico Azul Mónica Silva* (CADOICO), *Salgueira Branco do Mar* (SBM) and *Salgueira Branco Real* (SBR).

In this sense, we have access to a dataset of limestone slabs' images collected right after a polishing process. The limestone slabs are first fed into polishing equipment in order to eliminate the marks resulting from the abrasive cutting processes. Fig. 5 represents a polishing equipment, namely the *StonePOLISH* model, produced by CEI [23]. These slabs may have different dimensions, with a maximum of 3.5 meters in length and 2.5 in height. After this process, the plate is digitized by a scanner that reproduces the image as the plate moves on the carpet, saving the images on the polisher's own computer. These images are captured in an RGB color scheme and saved in .jpg format, and occupy between 1MB to 3MB of disk space; most have a size of 2800 by 1500. Also, all images are collected in one dataset (CADOICO, SBM, and

 ISBN: 978-1-61208-882-2

SBR), in a total of 954 copies. In this case, the classification of limestone categories is done manually, but there are efforts to automate this process [24]. From 954 images, 707 were labeled as normal and 247 as defective.

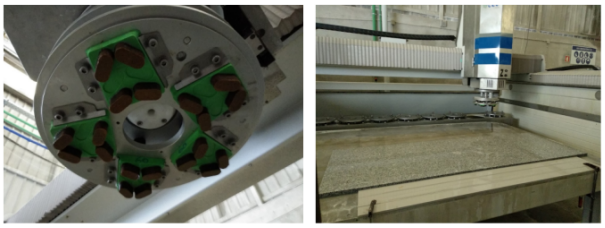

Fig. 5. Example of a polishing equipment.

Data augmentation was performed by applying rotations to the original images with 90º, 180º, and 270º, resulting in a total of 3816 images. This step was implemented to improve the CNN-based classifier performance, as data augmentation is widely recommended for image classification [25]. This augmented dataset was then split into train, validation, and test sets, with the same percentages mentioned before in Section III-B2. This division was needed for the CNN-based model, as only this classifier needs training and validation. For this training, an image input size of 400 was defined due to computational resources' limitations.

The proposed ensemble method was evaluated on the test set of the augmented limestone dataset mentioned before over standard performance metrics. From the three types of limestone mentioned, two levels of surface uniformity (calculated as detailed in Section III-A) were found: high homogeneity/uniformity for SBM and SBR (uniformity values bellow 9) and low uniformity/ high heterogeneity for CADOICO (uniformity values above 10). As the model's performance is highly impacted by this uniformity level, the performance metrics were retrieved for three variations of the dataset: (a) augmented homogeneous dataset (SBM and SBR augmented data); (b) augmented heterogeneous dataset (CADOICO augmented data); and (c) the complete augmented dataset (CADOICO, SBM, and SBR).

Also, to measure the performance improvement, the proposed model was compared with the isolated classifiers. A common metric like accuracy is suitable to assess the performance in defect detection but does not allows assessing if the classifier is good to distinct instances of *OK* and *NOK*. The ROC curve can be used for this, as it is a popular method for performance evaluation, plotting a graph with true positive rates over false positive rates. This way is possible to describe in a unique metric the trade-off between accurately classified positives and incorrectly classified negatives. The Area under the ROC curve (AUC) has been suggested as a robust classification performance metric, independent of the imbalance rate of the dataset, and can be used to compare performance between models completeness.

Given this, for validation of the classifier and comparison purposes, we have used AUC as the main evaluation metric.

TABLE I
AUC ON THE DIFFERENT AUGMENTED DATASETS WITH ADAPTIVE FILTERING, CNN-BASED, AND HYBRID WEIGHTED ENSEMBLE CLASSIFIER.

Datasets	**(a) Homogeneous**	**(b) Heterogeneous**	**(c) Complete**
	AUC	*AUC*	*AUC*
Adaptive Filtering-based	96.91%	84.84%	88.42%
CNN-based	93.3%	93.2%	95.8%
Hybrid Weighted Ensemble	96.96%	93.2%	96.04%

Table I shows the comparison of experimental results between each classifier alone (Adaptive filtering-based classifier and CNN-based classifier) and the proposed hybrid weighted ensemble method.

The AUC for an ideal and inaccurate model has values of 1 and 0.5, respectively. Given this, the average scores show that all the models perform accurately as well as exact and completeness. By comparing the AUC in the different variations of the dataset, it is clear that the adaptive filtering classifier works much better for homogeneous stones than for heterogeneous. This is mainly due to the classifier's difficulty with differentiating non-uniform patterns from defects, leading to false positives. Regarding the CNN-based model, although the AUC is worse than the filtering approach for the homogeneous stones, it is far more stable throughout all datasets, with less than 1% in AUC difference. When compared to the single models, the proposed ensemble model maintains the best performance from the isolated models, except in the total dataset where the AUC is increased.

The ensemble method's AUC in the homogeneous test stones outperforms the filtering model performance by a very small margin, indicating that the CNN-based model likely provided small betterment in classification when compared to the classification provided by the filtering approach alone. Conversely, in the heterogeneous dataset, the adaptive filtering model is not able to compensate for any CNN model's shortcomings, so the ensemble achieves the best possible result, the CNN's AUC. The ensemble model only shows significant improvements in the complete dataset. This is also where the applicability of this approach is more important. This increase in performance is derived from the compensation of the weak points of each isolated model.

As the weights given to each model's prediction derive from the stone's uniformity retrieved in pre-processing, more significance is given to the CNN in case of a low uniformity level and more to the filtering approach in case of a high uniformity value. This ensemble method also has the advantage of not utilizing extremely complex models that usually need more time in training and multiple parameter tuning, allowing for a lighter and simpler approach that achieves good performance while combining methods from different natures. By combining a supervised method with an adaptive filtering approach, the system is also more capable of correctly

 ISBN: 978-1-61208-882-2

inspecting new types of stones or defects that may appear in the future when compared to a typical supervised approach.

V. Conclusions and Future Work

Considering the ornamental stone industrial sector, the automated detection of process-induced defects in stone slabs during the polishing phase is a common and important problem. Currently, most of the polishing quality inspection is performed manually by human experts. This inspection is very subjective since it is based on aesthetic properties. We propose an automated monitoring system based on machine vision to assist human operators with the quality of a polishing process. This monitoring system relies on a weighted hybrid ensemble classifier, which classifies polished ornamental stone slabs as *NOK* or *NOK*. Tests were performed to classify the polishing quality in limestone slabs by using a dataset of images collected after a polishing process. By inspection of results, it is possible to conclude that the proposed approach outperforms isolated classifiers in the same conditions.

However, there are some limitations in this study. For starters, the proposed solution may not be considered a truly online monitoring system since tests were performed in a dataset of images, where ideally, they should be performed using images collected in real-time during the polishing process of stone slabs. Secondly, the dataset considered is limited to limestone slabs from a specific Portuguese region, which reduces the flexibility of an automated monitoring approach. Thirdly, the proposed approach considers only the classification of the stone slabs. It is out of scope the automated classification feedback for self-correction of the polishing process. Finally, the stone slab classification in this study does not consider some important parameters used manually for quality inspection, such as the final destination of the stone manufactured good and the requirements from a specific client in that product.

As future work, we will deploy and validate the solution in real polishing equipment for real-time monitoring. Also, considering all the possible aesthetic requirements related to the stone manufactured good, we are considering an active or reinforcement learning approach to be considered in the ensemble approach.

Acknowledgment

This research was supported by the project PRODUTECH-SIF - Solutions for the Industry of the Future, financed by the Portuguese National program COMPETE 2020 and by Portuguese funds through FCT – Fundação para a Ciência e a Tecnologia (UID/EMS/00712/2013).

References

[1] A. da Silva and I. Almeida, "Towards industry 4.0— a case study in ornamental stone sector," *Resources Policy*, vol. 67, p. 101672, 2020.

[2] G. Bonifazi and S. Marinelli, "Ornamental stone finished product aesthetic inspection and characterization through a digital spectrophotometric approach," in *Machine Vision Applications in Industrial Inspection XI*, vol. 5011. International Society for Optics and Photonics, 2003, pp. 243–250.

[3] J. G. Mathielo and T. M. Bolonini, "Diagnóstico do processo de polimento de rochas ornamentais," 2016.

[4] P. Nerakae, P. Uangpairoj, and K. Chamniprasart, "Using machine vision for flexible automatic assembly system," *Procedia Computer Science*, vol. 96, pp. 428–435, 2016.

[5] X. Su and Q. Zhang, "Dynamic 3d shape measurement method: a review," *Optics and Lasers in Engineering*, vol. 48, pp. 191–204, 2010.

[6] V. Montiel-Zafra, F. Canadas-Quesada, M. Campos-Suñol, P. Vera-Candeas, and N. Ruiz-Reyes, "Monitoring the internal quality of ornamental stone using impact-echo testing," *Applied Acoustics*, vol. 155, pp. 180–189, 2019.

[7] M. C. Proença, M. Aniceto, P. N. Santos, and J. C. Freitas, "Automatic detection of defects in ornamental limestone using wavelets," *International Journal of Computer and Information Engineering*, vol. 10, no. 5, pp. 864–869, 2016.

[8] G. Tantussi and M. Lanzetta, "Analyses of stone surfaces by optical methods," in *AI Te. M 2007, 8th Conference of the Italian Association of Mechanical Technology*, 2007, pp. 100–128.

[9] G. Bonifazi, A. Gargiulo, S. Serranti, and C. Raspi, "Imaging-based logics for ornamental stone quality chart definition," in *Machine Vision Applications in Industrial Inspection XV*, vol. 6503. International Society for Optics and Photonics, 2007, p. 65030P.

[10] R. Yarahmadi, R. Bagherpour, S.-G. Taherian, and L. M. Sousa, "A new quality factor for the building stone industry: a case study of stone blocks, slabs, and tiles," *Bulletin of Engineering Geology and the Environment*, vol. 78, no. 1, pp. 533–542, 2019.

[11] J. Lee, M. L. Smith, L. N. Smith, and P. S. Midha, "Robust and efficient automated detection of tooling defects in polished stone," *Computers in industry*, vol. 56, no. 8-9, pp. 787–801, 2005.

[12] F. Bianconi, E. González, A. Fernández, and S. A. Saetta, "Automatic classification of granite tiles through colour and texture features," *Expert Systems with Applications*, vol. 39, no. 12, pp. 11 212–11 218, 2012.

[13] J. Martínez, M. López, J. M. Matías, and J. Taboada, "Classifying slate tile quality using automated learning techniques," *Mathematical and Computer Modelling*, vol. 57, no. 7-8, pp. 1716–1721, 2013.

[14] C. Iglesias, J. Martínez, and J. Taboada, "Automated vision system for quality inspection of slate slabs," *Computers in Industry*, vol. 99, pp. 119–129, 2018.

[15] A. Ramil, A. López, J. Pozo-Antonio, and T. Rivas, "A computer vision system for identification of granite-forming minerals based on rgb data and artificial neural networks," *Measurement*, vol. 117, pp. 90–95, 2018.

[16] K. K. Pal and K. Sudeep, "Preprocessing for image classification by convolutional neural networks," in *2016 IEEE International Conference on Recent Trends in Electronics, Information & Communication Technology (RTEICT)*. IEEE, 2016, pp. 1778–1781.

[17] S. Suzuki *et al.*, "Topological structural analysis of digitized binary images by border following," *Computer vision, graphics, and image processing*, vol. 30, no. 1, pp. 32–46, 1985.

[18] H.-F. Ng, "Automatic thresholding for defect detection," *Pattern recognition letters*, vol. 27, no. 14, pp. 1644–1649, 2006.

[19] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger, "Densely connected convolutional networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 4700–4708.

[20] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, "Rethinking the inception architecture for computer vision," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 2818–2826.

[21] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

[22] J. M. Carvalho, J. V. Lisboa, A. Casal Moura, C. Carvalho, L. M. Sousa, and M. M. Leite, "Evaluation of the portuguese ornamental stone resources," in *Key Engineering Materials*, vol. 548. Trans Tech Publ, 2013, pp. 3–9.

[23] CEI, "Stone machinery - stonepolish," https://www.ceigroup.net/stonepolish, 2017.

[24] M. Tereso, L. Rato, and T. Gonçalves, "Automatic classification of ornamental stones using machine learning techniques a study applied to limestone," in *2020 15th Iberian Conference on Information Systems and Technologies (CISTI)*. IEEE, 2020, pp. 1–6.

[25] A. Mikołajczyk and M. Grochowski, "Data augmentation for improving deep learning in image classification problem," in *2018 international interdisciplinary PhD workshop (IIPhDW)*. IEEE, 2018, pp. 117–122.

 ISBN: 978-1-61208-882-2