Orientation Prediction in Robotics: A Study of Trigonometric Decomposition Methods Across Synthetic and Real-World Datasets

Antonio Gambale, Sonya Coleman, Dermot Kerr, Philip Vance
School of Computing, Engineering & Intelligent Systems,
Ulster University, Londonderry, Northern Ireland
e-mail: {gambale-a | sa.coleman | d.kerr | p.vance}@ulster.ac.uk

Emmett Kerr
Dept. of Electronic & Mechanical Engineering,
Atlantic Technological University, Letterkenny, Ireland
e-mail: emmett.kerr@atu.ie

Cornelia Fermüller, Yiannis Aloimonos
Institute for Advanced Computer Studies,
University of Maryland, USA
e-mail: {fermulcm | yiannis}@umd.edu

***Abstract*—Orientation prediction is a critical task for robotics as it enables robots to understand and interact with their environment more effectively. By accurately determining an object's position and orientation, robots can perform a range of complex tasks. This in turn will advance smart manufacturing facilities to achieve higher levels of automation, increase efficiency, and enable more flexible production systems. Hence, we present a comparative study of shallow regression models, integration strategies, and trigonometric encoding schemes for planar orientation prediction in robotics, using synthetic and real-world datasets. Results demonstrate that XGBoost 1.7, combined with vector integration and quadrant encoding, achieves the best balance of accuracy, robustness to angular boundary discontinuities, and computational efficiency, significantly outperforming alternative approaches in real-world scenarios.**

***Keywords*-Computer Vision; Robotics; Manipulation; Machine Learning; Smart Manufacturing.**

I. Introduction

Autonomous grasping serves as the foundation for numerous robotic operations, yet current approaches frequently generate arbitrary grasp poses without considering the object's pose for subsequent manipulation requirements [1] [2]. Contemporary robotic grasping research typically focuses on two-finger parallel grippers [1] [2], using complex seven or five dimensional grasp representations [3] [4], often referred to as grasping rectangles [5]. Some works have explored approaches to simplify grasp parameters whilst maintaining effectiveness. Notably, the work in [6] demonstrated success in representing object orientation using a single planer angle, predicted through ImageNet Convolutional Neural Network (CNN) feature extraction with a Support Vector Regressor (SVR) for angle prediction. However, the work demonstrated in [6] relies on RGB-Depth (RGB-D) data and complex hierarchical regression techniques to handle angle discontinuity at the 0°/360° boundary. This is accomplished using a two-step process: first, the angle is classified into one of four 90° intervals; then, a separate SVR model is trained for each interval. While effective, this method introduces additional complexity compared with single stage direct angle regression approaches that utilise RGB data. Additionally, alternative existing strategies present other notable drawbacks: methods that rely on high-dimensional grasp representations or hierarchical angle classification typically involve significant computational overhead, increased susceptibility to annotation errors, and limited generalisation when object pose varies beyond training data distributions. In contrast, approaches that require depth sensing or RGB-D inputs restrict applicability on platforms equipped only with RGB cameras. These constraints hinder practical deployment in dynamic or resource-constrained environments.

This research, therefore, aims to enhance post-grasp task execution through the use of a novel methodology which represents object orientation as a single angle in a 360° coordinate system. This approach provides orientation information that not only facilitates the intended use of the object but also enables derivation of the appropriate grasp angle. Additionally, this representation enables learning architectures to focus specifically on orientation prediction of one value (θ), circumventing the complexity of multiple parameter estimations and facilitating grasping with manipulators of all types.

To address the scarcity of annotated grasping datasets suitable for single angle object pose estimation, this work draws direct inspiration from [7], which utilised traditional computer vision with object geometric data to generate object poses. Here, a modified version of the methodology outlined in [7] is employed to augment the MetaGraspNet dataset: rather than relying on edge maps, ground-truth object segmentation masks are used to extract geometric features such as centroid, handle-tip displacement, and axis of symmetry, enabling the calculation of precise orientation annotations for each object instance. This automated annotation process was validated through manual inspection to ensure geometric fidelity and orientation accuracy, resulting in a high-quality synthetic dataset well-suited for model training. Recognising that synthetic datasets alone cannot capture the full complexity of real-world scenarios, a complementary real-world dataset was created and rigorously annotated. This secondary dataset was curated to reflect a range of object configurations, occlusions, and lighting conditions encountered in practical applications. The combination of

these two datasets (a large, systematically annotated synthetic set and a smaller, carefully validated real-world set) enables robust evaluation of model generalisation and domain transfer, providing a more comprehensive assessment of each method's practical viability. This research systematically evaluates the influence of regression architectures, integration strategies, and encoding schemes on orientation prediction performance. Both branched and vector-based integration approaches are assessed alongside various trigonometric encoding schemes, using synthetic and real-world data. Cross-validation, computational benchmarking, and error analysis are integrated to determine practical suitability, advancing the current state of planar orientation prediction for robotic manipulation. The study clarifies the interplay between encoding, integration, and model architecture, while also providing actionable insights for deploying robust, efficient orientation predictors in real-world robotic pipelines.

The remainder of this paper is organised as follows. Section II details the datasets used for training and evaluation. Section III describes the methodology, outlining the regression models, target encoding strategies, and integration approaches assessed in this work. Section IV presents the experimental results, including performance metrics and computational benchmarking. Section V provides a comprehensive discussion and evaluation of the findings, with a focus on model robustness, error distributions, and practical implications. Finally, Section VI concludes the paper and outlines future work.

II. Datasets

Training Dataset MetaGraspNet: The following work employs two distinct datasets: the first is used for training and validating the proposed methods, while the second serves to evaluate performance under conditions more closely aligned with real-world applications. The initial training of the proposed methods utilised a carefully curated subset of the MetaGraspNet dataset [8] reduced to focus exclusively on Phillips and flat-head screwdrivers. The subset selected for this work corresponds to the single-class, multiple-instance configuration within the MetaGraspNet framework. The choice of screwdrivers as the focal object class is deliberate as these objects possess geometric properties, which satisfy the requirements outlined in [7]: screwdrivers possess an axis of symmetry and a mass distribution that is biased to one side perpendicular to this axis. These characteristics permit the application of the approach described in [7] to compute the centroid (derived from the segmentation mask), the distances from the centroid to the screwdriver's handle and tip, and hence the identification of the direction based on the displacement of the centre of mass relative to the tip. This directional information is then recorded as an angular annotation, ranging from $0°$ to $359°$ and provides the angular information required to use this dataset to train machine learning models such as those employed in this paper.

The resulting curated dataset provides 7,932 annotations across 2,691 images of size 1200 x 1200 pixels [9]. Initial evaluation of the dataset revealed that orientation calculations for occluded objects with areas less than 10,000 pixels were unreliable. Consequently, such instances were removed, reducing the dataset to 5,709 annotations. A 10% sample of these annotations was manually verified, revealing 55 screwdrivers with incorrectly detected angles (errors ranging from $1°$ to $9°$), yielding a Mean Absolute Angular Error (MAAE) of $0.313°$ across the sample set. This inspection demonstrated that the majority of angular errors were minor and would not significantly compromise robotic grasping performance [10]. The final dataset thus comprises 4,567 training samples and 1,142 testing samples with associated images and annotations (see examples in Figure 1).

Figure 1. Example images from the MetaGraspNet training dataset.

Secondary Evaluation Dataset: A secondary real-world dataset, containing 81 annotations, was constructed to evaluate the trained models using real-world RGB images of screwdrivers. This dataset mirrors the structure of the curated MetaGraspNet subset employed for training, closely adhering to the difficulty levels defined in [9] but uses real screwdrivers rather than synthetic data. The MetaGraspNet difficulty levels are designed to progressively increase in complexity: Levels 1 and 2 represent single objects with no occlusion, while higher levels introduce multiple objects and increasing degrees of occlusion and clutter. In line with this, the custom real-world evaluation dataset contains 8 images at difficulty level 1 or 2 (single screwdrivers, no occlusion), 5 images at difficulty level 4 (moderate clutter and occlusion), and 14 images at difficulty level 5 (high clutter and occlusion). Example images from the evaluation set are shown in Figure 2. No images from difficulty level 3 were included, consistent with the original subset selection. Table I summarises the dataset details.

The secondary evaluation dataset was acquired using a camera equipped with a Samsung ISOCELL GN9 sensor, capturing images of size 3072 × 4080 pixels, with a lens aperture of f/1.9, an exposure time of 1/100 second, and an

International Organisation for Standardisation (ISO) sensitivity of 386, with images being collected from both overhead and 45° angled perspectives. The dataset features three screwdriver variants, red/black Torx, blue/black flathead/starhead, and solid black starhead, randomly arranged on a white backdrop under uncontrolled ambient lighting conditions. To ensure natural illumination variability, 27 images were taken across daylight hours, resulting in 81 annotations.

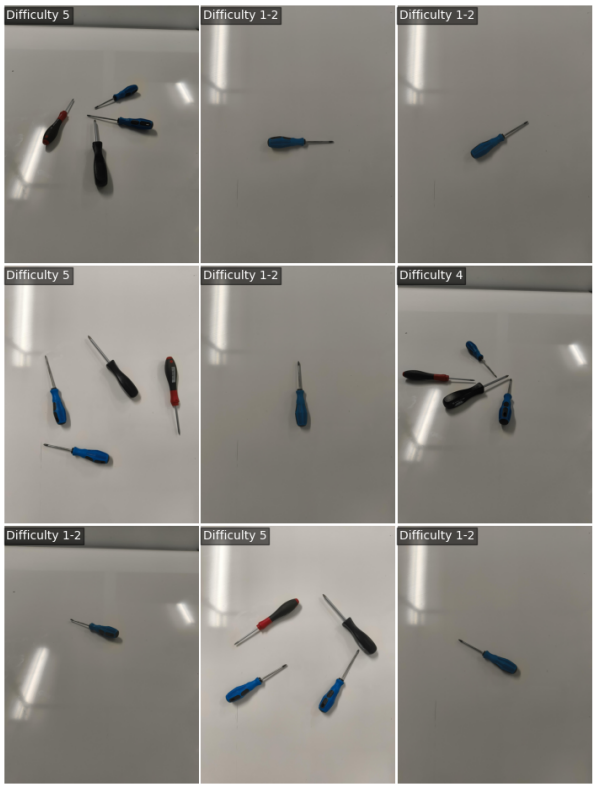

Figure 2. Example images from the real-world evaluation dataset.

Ground truth segmentation masks and orientation angles were annotated by three domain experts [11]. Inter-annotator agreement on segmentation masks was assessed using mean Intersection-over-Union (IoU), where IoU is calculated as the area of overlap between two masks divided by the area encompassed by either mask, providing a measure of agreement for object segmentation; this analysis yielded a high consistency of 0.95. Discrepancies were resolved via consensus review to refine the segmentation boundaries. For orientation, annotators measured shaft angles against the right horizontal axis, with a mean absolute difference of $\pm 1.8^\circ$ between predictions. Disagreements exceeding 5° were resolved by taking the circular mean for the affected objects; otherwise, the final orientation annotations were selected from a random sample of the independent measurements, except where consensus averaging was required as described.

TABLE I. SUMMARY OF IMAGES AND ANNOTATIONS BY DIFFICULTY LEVEL

Level	No. of Images	No. of Annotations
1-2	8	8
4	5	17
5	14	56

For inference in the custom real-world dataset, segmentation masks were generated using the Segment Anything Model (SAM) [12]. The accuracy of these masks was validated against the manual annotations, achieving a mean IoU of 0.95. This strong agreement supported the use of SAM-generated masks for subsequent stages in the pipeline.

III. METHODOLOGY

This work systematically evaluates planar orientation prediction using three regression architectures, two integration strategies, and four target encoding schemes. The methodology extends trigonometric decomposition to address circular data challenges and combines encoding scheme analysis with shallow learning models. Each combination's performance is assessed on both a MetaGraspNet subset and a custom real-world dataset.

A. Target Encoding Schemes

Four encoding configurations were designed, in which the data are derived from ground truth angles to form the study:

- **Base**: Fundamental trigonometric components:

$$\mathbf{y}_{\text{base}} = [\sin(\theta), \cos(\theta)]$$

- **Quadrant**: Base + one-hot quadrant encoding (Q1-Q4):

$$\mathbf{y}_{\text{quad}} = [\sin(\theta), \cos(\theta), \mathbf{1}_{Q1}, \mathbf{1}_{Q2}, \mathbf{1}_{Q3}, \mathbf{1}_{Q4}]$$

- **Polar**: Base + radian displacement:

$$\mathbf{y}_{\text{polar}} = [\sin(\theta), \cos(\theta), \theta_{\text{rad}}]$$

- **Full**: Comprehensive representation:

$$\mathbf{y}_{\text{full}} = [\sin(\theta), \cos(\theta), \mathbf{1}_{Q1}, \mathbf{1}_{Q2}, \mathbf{1}_{Q3}, \mathbf{1}_{Q4}, \theta_{\text{rad}}]$$

These encodings represent target variables used to train various shallow learning models, as outlined in Table II. The selection of these models is motivated by specific technical requirements of the orientation prediction task and practical deployment constraints. Random Forest (RF) was chosen for its native multi-output capability and robustness to noisy features, particularly relevant given the variable quality of geometric features extracted from segmentation masks. SVR was selected for its strong generalisation via kernel methods, enabling effective handling of non-linear relationships between ResNet50 features and trigonometric targets; Multiple-Output Support Vector Regression (M-SVR) extends this capability to joint optimisation across all target variables. XGBoost 1.7 implementations were included due to their established performance in regression tasks involving high-dimensional feature spaces and their gradient boosting approach's ability to iteratively correct prediction errors, a property especially valuable for circular data where small angular errors can compound. The XGBoost 2.0 variant specifically addresses multi-output limitations present in the earlier versions (1.7). Deep learning approaches were deliberately excluded due to the relatively modest dataset size (5,709 training samples),

which favours shallow models that avoid overfitting, as well as the need for rapid inference in robotic applications where computational efficiency is paramount.

TABLE II. SHALLOW LEARNING MODELS TESTED

Model	Description
RF	Random Forest
SVR	Support Vector Regression
M-SVR	Multiple-Output Support Vector Regression
XGBoost 1.7	Standard XGBoost implementation
XGBoost 2.0	Multi-output enabled XGBoost variant

Along with the multiple target encoding schemes and different model architectures, two proposed model integration strategies were introduced. This refers to how each of the models handles the target variables required to perform the analysis of encoding schemes. The two proposed integration strategies are as follows:

B. Branched integration strategy

This integration strategy uses multi-output models split into branches which are trained in parallel, one predicting the sine component and the other predicting the cosine component. In each of these branches, the additional target variables required for the encoding scheme analysis remain, meaning the only change is that the branch simply does not receive its alternative sin/cosine pair, but it does receive all other target variables required for the encoding scheme testing. During testing and subsequent inference, the predicted sine and cosine values are then combined using the inverse tangent function ($\tan^{-1}$) to provide the predicted angle.

C. Vector integration strategy

The second proposed integration strategy leverages a vector-based approach, however, depending on the model's architecture, the meaning of this differs slightly. For tree-based approaches, such as decision trees and RF, multiple outputs are natively supported, allowing a single model to predict all target variables directly, allowing for a joint relationship between targets sharing their influence on the scorers, loss function and predictions [13].

However, for algorithms lacking native multi-output support (e.g., SVR or XGBoost 1.7), a wrapper-based framework is required [13]. This wrapper fits one independent regressor per target variable, however it differentiates itself from the branched approaches by using shared hyperparameters tuned globally across all targets. This allows for a more balanced performance optimisation across all target variables through the use of a custom scorer that can be changed but does not alter individual model training objectives [14].

As noted previously, wrapper-based approaches face limitations when applied to models lacking native multi-output regression capabilities. To address this constraint and enable comprehensive comparative analysis, two additional implementations were evaluated: XGBoost 2.0 [15] and a M-SVR [16]. These vector-based approaches introduce critical architectural enhancements over standard wrapper methods. XGBoost 2.0 implements multi-output trees, where leaf nodes contain vector outputs spanning all targets simultaneously. This enables feature splits during tree construction to directly consider cross-target relationships, a capability absent when isolated models produced by Multi-Output Regressor wrappers. Concurrently, M-SVR extends traditional SVR by jointly optimising all targets through a unified ε-insensitive loss function, eliminating the hyperparameter compromises inherent in wrapper approaches. These additional models, when used with this integration strategy, should better enable implicit enforcement of trigonometric relationships (e.g., $\sin^2\theta + \cos^2\theta = 1$) more akin to the use of RF models.

D. Feature extraction

The pipeline (Figure 3) processes object segmentation masks (provided in MetaGraspNet annotations) to extract screwdriver patches. Each patch is placed on a white background, resized with aspect ratio preserved, and padded to 224×224 pixels. A ResNet50 CNN pre-trained on ImageNet, with classification head removed and global average pooling applied, extracts 2048 dimensional feature vectors [17]. These features serve as inputs to regression models trained on synthetic data from MetaGraspNet subset. For testing on real-world data, the same pipeline is applied using the custom dataset, with segmentation provided by SAM [12].

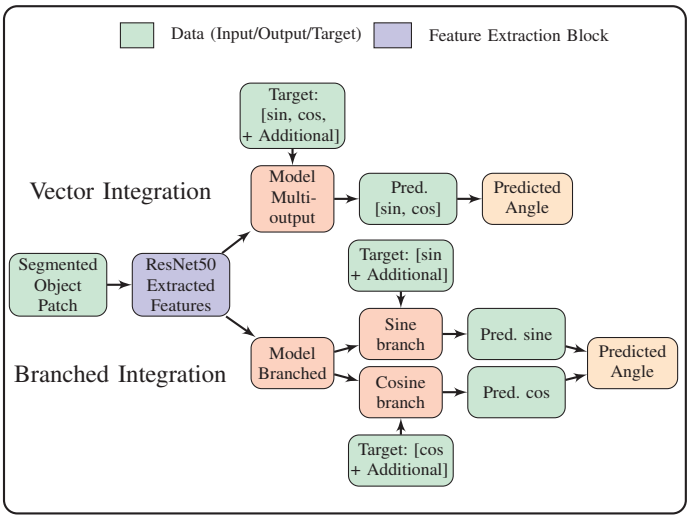

Figure 3. Object orientation prediction pipeline.

IV. RESULTS

To assess the predictive performance of the tested models, integration strategies, and encoding approaches, the MAAE was employed as the primary metric. This measure was computed independently for each encoding method and dataset, ensuring a granular evaluation of angular accuracy. Robustness was further enhanced through the application of 5-fold cross validation across all models and encoding scheme, providing a reliable estimate of performance.

A comprehensive computational benchmarking analysis was also conducted to evaluate inference speed. For each model, inference was performed on identical image features, with each test comprising 1000 repetitions per run and five aggregate

runs to account for variability. To ensure fairness, the sequence of model evaluation was randomised for each aggregate run, and system memory was cleared between measurements to minimise interference. All experiments were executed on a Linux 6.8.0 system equipped with Python 3.8.3, an Intel Xeon w7-3445 CPU (20 cores, 40 threads), and 62.3 GB of RAM. This rigorous protocol facilitates a thorough and equitable comparison of both predictive accuracy and computational efficiency across all models and integration strategies.

The following results begin with the MAAE measurements across the MetaGraspNet subset. The results compare different integration strategies (vector vs branched encoding) and encoding configurations (full target set vs reduced sets). All values represent degrees of angular error, with lower values indicating better performance. The initial results on the MetaGraspNet subset are displayed in Table III.

TABLE III. MAAE FOR METAGRASPNET DATASET (DEGREES)

Model	Base	Quadrant	Polar	Full
XGBoost 1.7 (Vector)	5.15	5.15	5.15	5.15
XGBoost 1.7 (Branched)	5.15	5.15	5.15	5.15
XGBoost 2 (Vector)	**4.92**	**5.01**	5.46	5.17
M-SVR (Vector)	8.04	8.04	8.04	8.04
SVR (Vector)	5.01	5.12	**5.01**	**5.01**
SVR (Branched)	5.01	5.12	**5.01**	**5.01**
RF (Vector)	5.48	**5.08**	5.87	5.67
RF (Branched)	7.43	5.44	5.93	5.88

TABLE IV. MAAE RESULTS FOR REAL-WORLD DATASET (DEGREES)

Model	Base	Quadrant	Polar	Full
XGBoost 1.7 (Vector)	**9.61**	**8.15**	**8.96**	**9.61**
XGBoost 1.7 (Branched)	**9.61**	11.66	**9.15**	11.14
XGBoost 2 (Vector)	17.09	13.91	**7.18**	10.46
M-SVR (Vector)	28.86	28.86	28.82	28.82
SVR (Vector)	23.13	23.54	23.14	28.03
SVR (Branched)	23.15	23.54	23.14	23.28
RF (Vector)	14.81	14.49	11.84	17.43
RF (Branched)	16.45	**11.62**	12.90	17.50

TABLE V. INFERENCE TIME PER IMAGE PATCH (MILLISECONDS)

Model Type	Base	Quadrant	Polar	Full
XGBoost 1.7 (Vector)	0.76	1.86	0.91	1.73
XGBoost 1.7 (Branched)	0.50	0.83	1.20	3.61
XGBoost 2 (Vector)	0.76	1.86	0.91	0.29
M-SVR (Vector)	17.76	17.78	17.80	17.76
SVR (Vector)	13.48	41.94	20.78	51.25
SVR (Branched)	13.55	70.23	27.68	75.49
RF (Vector)	59.27	56.50	57.90	45.42
RF (Branched)	119.15	117.08	117.83	91.04

Table IV summarises model performance on the real-world dataset using the same MAAE metric to enable direct comparison with the MetaGraspNet results. In addition to angular accuracy, Table V reports the inference time per image (in milliseconds) for each model and condition. This enables an explicit comparison of computational efficiency across integration strategies and model types, with lower values reflecting faster processing.

V. Discussion | Evaluation

While the MAAE metrics presented in Section IV form the core of this discussion, it is essential to first address the primary challenge that this work seeks to overcome: the issue of boundary discontinuity. To thoroughly assess each method's performance, we conduct a granular analysis of the inference data on the custom dataset, as visualised in Figure 4. Figure 4 displays the prediction error as a function of the ground truth angle for each model and encoding configuration. Specifically, the x-axis represents the ground truth angle of the object, while the y-axis shows the signed angular error, calculated as the shortest difference between the predicted and actual angles, wrapped to the interval $[-180°, 180°]$. Each point corresponds to a single prediction, positioned horizontally by its ground truth angle and vertically by the deviation from the true value.

A detailed inspection of these plots reveals substantial differences in error distributions between models, which are not always reflected in the aggregate MAAE values. For instance, the M-SVR model exhibits pronounced and frequent error spikes at the $0°$ and $359°$ boundaries, indicating a persistent struggle with boundary discontinuity and a lack of robustness in these critical regions. These errors are not isolated; the M-SVR model demonstrates erratic behaviour across much of the angular range, with large, abrupt deviations that suggest poor generalisation and reliability. In contrast, XGBoost 1.7 (both vector and branched variants) stands out for its consistent and stable error profile. Across nearly the entire angular range, prediction errors remain tightly clustered around zero, with only occasional moderate spikes, mostly at angular boundaries. This stability is indicative of a model that not only achieves a low mean error, as seen in Table V, but also avoids catastrophic failures, making it more suitable for real-world deployment where reliability is paramount.

XGBoost 2, while achieving competitive MAAE values in some configurations, displays a more volatile error pattern. Notably, it exhibits significant errors not only at the $0°$ and $359°$ boundaries but also around $180°$, suggesting that its generalisation may be compromised at multiple critical angles. This behaviour underscores the importance of evaluating models beyond mean metrics, as a low MAAE can mask underlying instability. The SVR models, both vector and branched, provide mixed results. While their errors are generally moderate, both models are prone to sporadic, large prediction failures at various angles, particularly near the boundaries and occasionally in the mid-range. These large spikes indicate that, although SVR may perform adequately on average, it is susceptible to unpredictable outliers that could undermine its practical utility. Random Forest models show moderate stability, with the vector variant generally outperforming the branched version in terms of error consistency. While occasional spikes are present, these are less frequent and less severe than those observed in M-SVR or

SVR, positioning Random Forest as a reasonable compromise between stability and accuracy, something reflected by its relatively low MAAE when paired with quadrant encoding and a branched integration strategy as seen in Table V.

It is important to note that all models were trained exclusively on a synthetic dataset which is inherently simpler and more controlled than the real-world test set. As a result, the MAAE values achieved using the synthetic MetaGraspNet subset (Table III) are substantially lower across all models, reflecting the training-domain familiarity. For example, XGBoost 1.7 achieves a MAAE of 5.15° using the synthetic data, compared with 8.15–9.61° using the real-world dataset, while M-SVR and SVR also show marked increases in error when transitioning to real-world evaluation. This domain gap highlights the challenge of generalising from synthetic to real data, and underscores the value of robust error analysis using the real-world test set.

In addition to predictive accuracy, computational efficiency was also evaluated. Table V quantifies inference times per image patch, demonstrating XGBoost 1.7's superior performance: configurations predominantly achieve sub-2ms inference times (one outlier). In contrast, M-SVR/SVR models exhibit 10-100× slower performance (13.48-75.49ms), while RF demonstrates the poorest efficiency, consistently exceeding 45ms and frequently surpassing 100ms. This efficiency advantage positions XGBoost 1.7 as optimal for applications requiring both rapid inference and angular reliability.

Taken together, these results demonstrate that XGBoost 1.7 provides the most robust and reliable predictions across the full angular range using real-world data, effectively managing boundary discontinuities and avoiding large, erratic errors, while also offering leading computational efficiency. In contrast, models such as M-SVR and SVR are hindered by frequent and severe outliers, particularly at critical boundaries, and Random Forest occupies a middle ground, providing reasonable stability but not matching the overall reliability or speed of XGBoost 1.7. This comprehensive evaluation highlights the necessity of considering both aggregate metrics and detailed error distributions, as well as computational efficiency, when selecting models for applications where consistent and timely performance across all angles is essential.

A key focus of this work is the interplay between encoding strategies, model architectures, and integration approaches. Therefore, these results also reveal that model responsiveness to encoding varies considerably:

- **XGBoost 1.7** demonstrates strong robustness across all encodings, but achieves its best real-world performance with quadrant encoding in the vector integration configuration, yielding the lowest MAAE (8.15°). Notably, the addition of quadrant or polar information generally improves performance over the base (sin, cos) encoding, suggesting that XGBoost 1.7 is able to leverage richer target representations to better manage boundary effects and reduce systematic errors. The full encoding does not consistently outperform quadrant or polar, indicating that the full encoding may introduce redundancy or noise for this architecture.

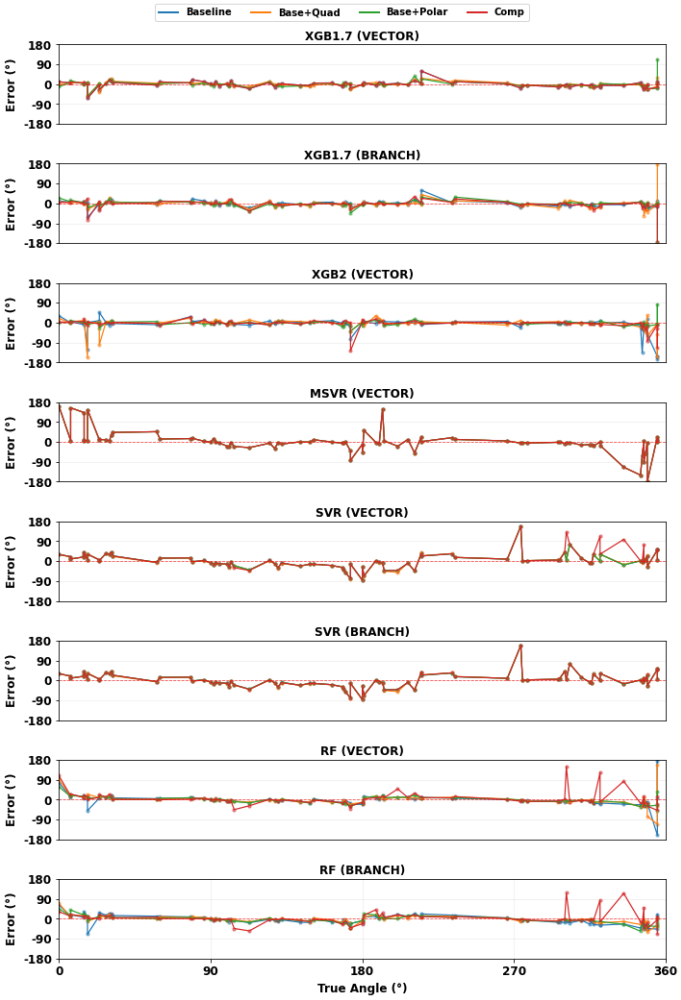

Figure 4. Model prediction error vs ground truth angle on real-world test set (missing data in 355° and 359° range).

- **XGBoost 2.0** is more sensitive to the encoding choice, achieving its best real-world MAAE (7.18°) with the polar encoding. However, its error distribution is less stable, with notable spikes at both boundary and mid-range angles, suggesting that, while certain encodings can lower mean error, they may not guarantee robust predictions.
- **SVR and M-SVR** models show limited benefit from more complex encodings. Both models exhibit high MAAE and frequent large errors regardless of encoding, indicating that their architectures are less capable of exploiting additional target information to improve generalisation, especially in the presence of boundary discontinuities.
- **Random Forest** benefits moderately from quadrant encoding, particularly in the branched configuration, but does not reach the accuracy or stability of XGBoost models. Its performance is more consistent than SVR/M-SVR but less robust to encoding changes than XGBoost 1.7.

The choice between branched and vector integration strategies also influences model performance:

- For **XGBoost 1.7**, the vector strategy paired with quadrant

encoding is optimal, while the branched approach is less effective, especially when combined with more complex encodings.

- **Random Forest** shows a preference for the branched strategy when used with quadrant encoding, achieving its lowest MAAE (11.62°), but remains slower and less accurate than XGBoost.
- **SVR** and **M-SVR** do not benefit significantly from either integration strategy, with both approaches yielding high error and erratic predictions.

In summary, these results demonstrate that XGBoost 1.7, with vector integration and quadrant encoding, offers the best trade-off between accuracy, robustness to boundary discontinuities, and computational efficiency using real-world data. While more complex encodings marginally improve mean error for certain models, the quadrant encoding strikes an effective balance between informativeness and generalisation. Models such as M-SVR and SVR are hindered by frequent and severe outliers and do not benefit meaningfully from richer encodings or alternative integration strategies. Random Forest provides moderate stability but cannot match the overall reliability or speed of XGBoost 1.7. These findings underscore the need to consider both aggregate metrics and error distributions, alongside computational efficiency, when choosing models for consistent, timely performance.

VI. Conclusion and Future Work

This work systematically evaluates a range of shallow regression models, integration strategies, and target encoding schemes for the challenging task of planar orientation prediction using both synthetic and real-world datasets. The results demonstrate that among the tested configurations, **XGBoost 1.7**, when paired with the **vector integration strategy** and **quadrant encoding**, consistently delivers the best trade-off between predictive accuracy, robustness to angular boundary discontinuities, and computational efficiency on real-world data. While more complex encodings, such as the full or polar representations, can marginally improve mean error for some models, the quadrant encoding achieves a superior balance between informativeness and generalisation, avoiding the pitfalls of overfitting or redundancy. Building on the insights gained from this study, future work will prioritise scaling to larger and more diverse datasets, extending evaluation to complex object categories, and exploring deep learning and hybrid approaches to benchmark gains over shallow architectures. Finally, efforts will focus on full pipeline integration and real-world robotic deployment, enabling end-to-end assessment in closed-loop manipulation scenarios.

References

[1] X. Gao and Z. Wen, "Task-oriented robotic grasping for intelligent manufacturing", in *2023 3rd International Symposium on Artificial Intelligence and Intelligent Manufacturing (AIIM)*, IEEE, 2023, pp. 101–104.

[2] H. Sekkat, O. Moutik, L. Ourabah, B. ElKari, Y. Chaibi, and T. A. Tchakoucht, "Review of reinforcement learning for robotic grasping: Analysis and recommendations", *Statistics, Optimization & Information Computing*, vol. 12, no. 2, pp. 571–601, 2024.

[3] L. Chen, P. Huang, Y. Li, and Z. Meng, "Detecting graspable rectangles of objects in robotic grasping", *International Journal of Control, Automation and Systems*, vol. 18, no. 5, pp. 1343–1352, 2020.

[4] J. Redmon and A. Angelova, "Real-time grasp detection using convolutional neural networks", in *2015 IEEE international conference on robotics and automation (ICRA)*, IEEE, 2015, pp. 1316–1322.

[5] S. Kumra and C. Kanan, "Robotic grasp detection using deep convolutional neural networks", in *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, IEEE, 2017, pp. 769–776.

[6] M. Schwarz, H. Schulz, and S. Behnke, "RGB-D object recognition and pose estimation based on pre-trained convolutional neural network features", in *2015 IEEE International Conference on Robotics and Automation (ICRA)*, 2015, pp. 1329–1335. DOI: 10.1109/ICRA.2015.7139363.

[7] A. Gambale *et al.*, "A comparative study of hough transform and pca for bolt orientation detection", in *2024 IEEE 22nd International Conference on Industrial Informatics (INDIN)*, IEEE, 2024, pp. 1–6.

[8] Y. Chen, E. Z. Zeng, M. Gilles, and A. Wong, "Metagraspnet_v0: A large-scale benchmark dataset for vision-driven robotic grasping via physics-based metaverse synthesis", *arXiv preprint arXiv:2112.14663*, 2021.

[9] M. Gilles, Y. Chen, T. R. Winter, E. Z. Zeng, and A. Wong, "Metagraspnet: A large-scale benchmark dataset for scene-aware ambidextrous bin picking via physics-based metaverse synthesis", in *2022 IEEE 18th International Conference on Automation Science and Engineering (CASE)*, IEEE, 2022, pp. 220–227.

[10] A. Gambale *et al.*, "Computing the orientation of hardware components from images using traditional computer vision methods", *Engineering Proceedings*, vol. 65, no. 1, p. 8, 2024.

[11] A. Gambale, *"Screwdrivers dataset"*, https://app.roboflow.com/mvtec-uihve/screwdrivers-kmij6/2 (accessed: 2025-06-13), 2025.

[12] A. Kirillov *et al.*, "Segment anything", in *Proceedings of the IEEE/CVF international conference on computer vision*, 2023, pp. 4015–4026.

[13] J. Brownlee, *Ensemble Learning Algorithms With Python: Make Better Predictions with Bagging, Boosting, and Stacking*. Machine Learning Mastery, 2021.

[14] H. Borchani, G. Varando, C. Bielza, and P. Larranaga, "A survey on multi-output regression", *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, vol. 5, no. 5, pp. 216–233, 2015.

[15] T. Chen and C. Guestrin, *XGBoost 2.0: Scalable tree boosting system*, https://xgboosting.com/xgboost-2.0/ (accessed: 2025-06-13), 2024.

[16] Y. Bao, T. Xiong, and Z. Hu, "Multi-step-ahead time series prediction using multiple-output support vector regression", *Neurocomputing*, vol. 129, pp. 482–493, 2014, ISSN: 0925-2312. DOI: https://doi.org/10.1016/j.neucom.2013.09.010.

[17] A. Depierre, E. Dellandréa, and L. Chen, "Scoring graspability based on grasp regression for better grasp prediction", in *2021 IEEE international conference on robotics and automation (ICRA)*, IEEE, 2021, pp. 4370–4376.