Leveraging Gamma Corrections for an Overhead Reduced Mood Adaptive Display Coloring

Lukas Brodschelm, Felix Gräber, Daniel Hieber and Marc Hermann

Dept. of Computer Science
Aalen University
Aalen, Germany
Email: firstname.lastname@hs-aalen.de

***Abstract*—Humans can recognize a wide range of colors and interpret them in many different ways. Besides obvious effects like highlight and beautification, these colors can influence the emotional state of humans in a significant way. While this is no new information and color psychology is a heavily discussed topic in the psychological area, little research has been conducted in the human-computer interaction area of this topic. We presented a mood adaptive display coloring prototype in a previous paper, in this work the second iteration of the prototype is introduced. The second version extends the previous implementation, which utilizes psychological studies and state of the art machine learning technologies, with a new gamma shift-based coloring approach we present a greatly overhead reduced coloring service. The focus of using gamma shifting instead of overlays is to optimize the software for use on low-performance computers. To obtain equivalent results an approach is introduced, how to calculate a gamma-based shift, that is similar to alpha blend overlays. In order, that the overhead reduced version is still able to have the same influence effects on the emotion of human users as the previously presented prototype.**

Keywords*—Advanced Human Computer Interaction; Mood Adapting Coloring; Adaptive Display Coloring; Color Psychology; Emotion Recognition.

I. INTRODUCTION

Colors have been an important part of Human-Computer-Interaction (HCI) since the introduction of multi color displays. Nowadays, it is impossible to imagine software without them, the old default black and white is only used as a stylistic instrument. Even hardware is often produced with ambient lighting as a selling point. Unconsciously perceived colors, like ambient lights or screen taints, can have impact on the users. In our previous work, we introduced a novel approach using these effects to influence the users' mood [1].

Colors are used as an important part of designs for Graphical User Interfaces (GUIs) in software like games, operating systems (OS) and business applications, either to help users to get the required overview or leading workers through complex procedures with intelligent coloring.

However, the possibilities of colors do not end with highlighting certain things and improving the quality of life; therefore, they can also be used to directly affect the user. Understanding the influence of color on the human user, clever coloring can also be used to change the users attitude, e.g., to get the users trust on a website [2].

That colors can have a quantifiable effect on the human behavior on special aspects is claimed in several publications [2]–[5]. The effect of colors on the mood of a person have also been studied but studies have not yielded clear results [6]. This is why color psychology is not only a highly disputed topic but also an an arguable research area. A possible reason for this might be, that the association and effect of color highly depends on the subject's background not only age and gender but also cultural aspects and probably many more have to be considered [3], [7]–[9].

However, thanks to the popularity of the topic much research is conducted in this area, allowing a solid foundation for further research in the HCI environment. Depending on this knowledge we introduced an approach to interact with users through decent adaptive coloring, providing emotional support and the best conditions to successfully master their current task in a preceding paper [1]. The software we developed for this is called MAD Coloring (Mood Adaptive Display Coloring), it is a display overlay framework, which taints the display color using an overlay, that is applied to the screen using an alpha blend algorithm.

The provided MAD prototype exposes a generic API for clients to control the color according to the mood of the user. Further two simple clients were implemented. One of them provides a minimal boilerplate for further implementations, the second one utilizes state of the art machine learning techniques to recognize mood changes and provide the best suited coloring solution for the detected mood state.

Using state of the art technologies, like facial recognition systems or intelligent devices such as smartwatches, the mood of a user can already be recognized and measured quite precisely. The resulting information can be used to classify the emotions and start processes to reinforcing or combating these.

In this paper, we introduced an alternative implementation to our last Coloring Service, replacing the overhead heavy Hudkit service with an overhead reduced gamma shift implementation. This includes, but is not limited to an updated configuration file handling, the connection to the X.Org display server and a transformation from RGBA based alpha blend to gamma correction, that ensures compatibility with already existing configuration files. Furthermore, we show the differences between the first, alpha blend overlay and the gamma based version.

The remainder of this paper is structured as follows: Section

II provides background information for a better understanding of the work and introducing required secondary research. Section III highlights the general concept of MAD-Coloring, while Section IV explains the currently implemented state of MAD-Coloring. Following Section V compares the newly introduced gamma shift Coloring Service with the old Hudkit service. In Section VI we introduce a possible Case Study to empirically evaluate our prototype. Section VII provides an overview of related works. Finally, Section VIII concludes the paper and lists further research options.

II. Background

This section outlines the elementary information about color psychology and display coloring frameworks, as well as face recognition basics and a survey about the emotional effect of color, conducted during this work.

A. Color Theory and Psychology

While color is often accepted as an omnipresent thing without further questioning, it is a well-defined construct which can be described accurately with its three components: hue, value, and chroma [10]. The emotional effect of color is closely linked to all three of those; therefore, an adjustment of one of its values can lead to a completely different emotional reaction. This means, e.g., a green with high brightness and saturation has a different emotional effect than a green with the same hue and saturation but low brightness.

Even though color psychology is highly disputed down to its fundamental theories, it is possible to elaborate some general statements that are used in this work. While many animals only see different shades of grey and some can even see with infrared or ultraviolet vision, humans have settled in the middle with complex color perception. This has lead to the development of a society where colors play an important role in the day to day life, religion, work, and free time. Most people have a favorite color, and every kid, regardless of their origin, knows the red cross, star or moon provides them with help if needed.

While most humans see color the same way, it has quite different meanings for them. The context in which a color is experienced and from whom is fundamental for its interpretation. Age is an important factor, as studies prove, that children have different associations with certain colors than adults or elder people. Gender can also play a decisive role [8][3][9].

Elliot and Maier laid the theoretical base for contextual color psychology with their work in 2012 [8], highlighting six key properties of the psychological impacts of color:

- There might be psychological relevant associations with a color.
- Presentation colors might influence psychological operations, including but not limited to basic impulses such as attraction and avoidance.
- Associations with colors might trigger affective, cognitive or behavioral reactions. This happens subconsciously or without intention.
- Color meanings and associations are influenced by both trained and inherited behavior.
- The relation between color perception and association is bidirectional. color perception has an impact on psychological processes and psychological processes have an impact on the way color is perceived.
- The psychological effect of color heavily depends on the context. The context is so important, that the influence of the same color might result in opposite effects for different contexts.

Those core statements are used by other research (e.g., [3]) in the field of color psychology and should be considered when targeting a psychological effect utilizing colors.

Li [9] gives a detailed overview of the preferences and effects with different groups of people in a medical context, more precisely during hospital stays. Children prefer brightly saturated colors and overall very colorful environments. These provide distraction from the tense situation and a calming effect, connected to the coloring, could be proven. Adults on the other hand tend to favor clean, cool colors like white, blue, and grey. They associate these with a professional environment and thus a higher chance of successful treatment. There is also a connected calming effect. However, the effect is achieved through the impression of a professional environment, that evokes the feeling of a qualified treatment when seeking medical care. Therefore, it is only partially connected to the color and already an effect of the original feeling emitted by the color: professionalism. The author further states, that elder people tend to prefer warmer colors, although not as bright and less saturated than children. These colors help them to relax and effectively reduce anxiety. The professional, cool colors like white or grey even provide a negative effect on elders and sometimes children. This originates from the partly occurring associated with anxiety, loneliness, and fear. The phenomenon is known in color psychology as ”white scare”.

Focusing on the target-context oriented effect of color, Maier et al. outline how the psychological impact of color changes with the task domain [5]. Stated are the findings of different studies which point out that colors provide a performance-enhancing effect in physical competitions. An example of this can be found in sports, where teams wearing red tricots win more games than those wearing any other hue. However, when it comes to an intellectual target-context the color red seems to have no, or even a negative effect, on the performance.

B. Emotion Recognition

Facial and Image Recognition is currently one of the most popular fields of machine learning. Elementary face detection is not a technical challenge anymore and a camera input can be analyzed in realtime, with only a few dozen lines of code (e.g., [11]).

The popularity of such use cases provides us with the base for emotion recognition through feature detection in the mimic of people's faces. Combining this technology with display coloring frameworks allows us to design systems that are capable of autonomously detecting emotions and correctly using different color overlays to improve and reinforce them.

Several models have been trained with fairly high accuracy to detect emotion from images. These models focus on a combination of facial features to detect the emotional state behind pictures of faces. The model of Yang et al. [12] relies on the extraction of the mouth and eyes, achieving successful detection rates of up to 93% for certain emotions and a mean of 87%.

As emotions have a broad impact on the users' behavior the detection is not only limited to image recognition. For precise detection of emotions, multiple sources of information can be used. Ghosh et al. suggest an approach to determine emotion based on speech recognition [13]. By combining multiple ways of emotion recognition detection accuracy can be improved even further. However, background software with recording capabilities is rather problematic from a privacy point of view. Therefore, the improved recognition accuracy does not outweigh the privacy violation.

Other possibilities to increase the accuracy and therefore the value of automatic emotion detection could be provided by smart wearables like smartwatches or fitness trackers. Measurements like the pulse could then be used to determine the stress level and provide other valuable insights. However, as the prototype should be as lean as possible this approach is not further considered in this work. The required multi machine solution with means of communication between devices would cause a too excessive codebase for the aspired simple prototype.

C. Emotional Effects of Colors - a Survey

As our excessive study of related research projects and studies could not provide a common interpretation of the concrete connection between emotions and colors, a simple survey was conducted. This survey, however, is only used to generate a default profile for the prototype, it is not scientifically representative. In the survey, 522 people were asked to think of color if confronted with an emotion. Multiple answers were allowed, this should allow the detection of patterns, e.g., all warm colors are connected with emotion X. The emotions which could be chosen in the survey were (Ca)lm, (V)italising, (Sa)fety, (Co)ncentrated, (M)elancholy, (St)ressed and (H)appieness. The possible answers consisted of colors which can easily be displayed on a display and provide strong contrasts to each other, allowing a meaningful implementation.

TABLE I. EMOTIONAL CONNECTION TO COLORS SURVEY, RESPONDENTS = 522, VALUES IN %

Color	Ca	V	Sa	Co	M	St	H
red	3,79	14,86	13,38	7,32	9,37	**35,20**	8,45
orange	8,24	**19,41**	**18,09**	6,02	3,02	16,94	13,49
yellow	3,53	**21,15**	7,21	8,95	3,78	14,31	17,53
green	**22,09**	**18,34**	**15,44**	15,49	2,27	2,96	**24,72**
blue	**25,23**	9,64	**16,32**	**28,74**	7,55	4,77	14,25
violet	11,37	3,35	11,32	5,34	11,48	5,76	6,56
grey	11,11	0,40	5,44	13,08	**27,49**	3,95	2,27
black	12,81	2,68	10,29	13,43	**29,61**	6,74	4,41
pink	1,83	10,17	2,50	1,72	5,44	9,38	8,32

The survey's findings match well with the general statements from Rider [3]. While warmer colors (orange, yellow) have an arousing effect, cooler colors (blue, green) have a relaxing, calming effect. Rider further mentions the widespread of possible emotions connected to green depending on brightness, saturation, and context. As we only asked for the hue the subjects could not differentiate between different types of green. Therefore, green is strongly connected with the most positive emotions, which, however, most likely refer to different types of green.

The survey results also concur with the findings of Kido [4], reinforcing the calming effect of green and blue. The deafening effect of red, determined by Kido, also matches the very strong association with stress by our survey.

The strong association of orange with safety, which cannot be found in other studies or surveys, most likely is caused by a bias of our survey. The poll was conducted on a group of people from the same community, where orange is a frequent and positively interpreted color, which could explain this behavior.

D. Display Overlays

Display overlays are no new research topic and many refined products are available. In the gaming area, they are omnipresent with the most popular example being steam's in-game-overlay, other popular examples being the Nvidia Geforce Experience overlay, the Discord overlay or Overwolf. However, these are all limited to the gaming use cases and only Overwolf allows great modification freedom.

Overlays might either display solid colors or transparent graphics on top of other applications. When using unicolored transparent images, defined by a red, green, blue, and alpha channel (RGBA), the screen is tinted with the defined color. This procedure, of combining two images, is known as alpha blending or alpha compositing [14]. However, there is not just one but many ways to combine two images. Therefore, there are several formulas, which might be used to calculate the resulting color for the alpha blend.

The pqiv image viewer can display transparent pictures and can easily be placed on top of other windows [15]. Due to its implementation in Python, cross platform capabilities are provided. However, the lack of click-through support requires extensive extension work to create a functional, non-blocking overlay.

OnTopReplica, a C# project, supports the display of selected windows on top of others on Windows systems [16]. While it supports click-through and adjustable opacity, it is limited to Windows and requires an additional overlay-window. This not only decreases the compatible systems but also increases the overhead significantly.

Hudkit is a C based framework with an exposed JavaScript-API [17]. The framework supports all Linux X-desktops and some OS X systems. Its main HTML page can be modified like

Overwolf overlays using HTML and JavaScript. The page can then receive new input via established APIs like Websocket or WebGL and change the display accordingly. Multiple monitors and click-through events are supported by default. Providing a powerful small footprint framework.

E. Gamma Correction

Gamma correction is a luminance adjustment technique, designed to provide color genuine images on common computer monitors. The gamma(γ), which is added to the luminance of the displayed image relies on an exponential scale, the resulting color value is usually calculated for every pixel using the formula from Equation (1) (cf. [18] Advanced Lighting → Gamma Correction).

$$\vec{C}_{RGB} = \begin{pmatrix} Pixel_{red}^{\frac{1}{\gamma_R}} \\ Pixel_{green}^{\frac{1}{\gamma_G}} \\ Pixel_{blue}^{\frac{1}{\gamma_B}} \end{pmatrix} \tag{1}$$

Where gamma might be any floating point value between 0 and 10. This exponential approach is chosen as the human eye senses changes in the lower brightness spectrum more than those for bright colors (cf. Foley et al. [19] pp. 448-449).

Gamma corrections are freely available to the user on the major operating systems. On Microsoft Windows via DXGI [20] and on Linux Systems via the X.Org display service utilizing APIs like Xlib [21]. Apple's macOS also provides some methods to manipulate gamma, however, they lack sufficient public documentation. The gamma correction is configured within the display settings and is applied to the visible screen by the display service or operating system without the need to run any additional third party software.

On X.Org based Linux desktops this gamma correction might be configured via the Xlib API or one of its corresponding bindings, providing programmatic access to the X11 display service. However, the library interface itself is quite extensive and tools like Xrandr, which provides the functionality of the Xlib and its libXrandr extension as a command line tool provide a much more simplified interface [22].

Xrandr allows the user to adjust the gamma correction for every color channel via the command line option: -gamma R:G:B. By applying values larger than 1 the luminance is increased, by applying smaller values the luminance is decreased. When setting different luminance values for the different channels a color shift is applied to the screen.

III. Concept

This section provides insights into the architecture and concept for the current MAD-Coloring prototype. The modular architecture approach is shown in Fig. 1.

The architecture is separated in 3 sections. The Coloring Service, the clients accessing and controlling the coloring and the interface, enabling communications between the clients and the display coloring.

The display coloring has to be able to project a color adjustable overlay on top of all connected monitors. While we stated in our previous work [1] that gamma corrections are not sufficient we have to stand corrected. While linear gamma or brightness adjustments are not enough to fully leverage the psychological effects of the overlay, a calculated vector based gamma shift as introduced in this paper is capable of producing promising results. The display must not reduce the productivity and therefore has to enable all actions which are possible without an overlay, e.g., right/left clicking or text selection. Further, the vision quality and readability of displayed content must not be reduced any further than necessary. This will be compared against the light constraints coming with night lights, e.g., reduced contrast and slightly changed colors. The Coloring Service further gets supplied with at least one user profile file. This allows users to adjust the color optimally for themselves and maximize the effect of MAD-Coloring.

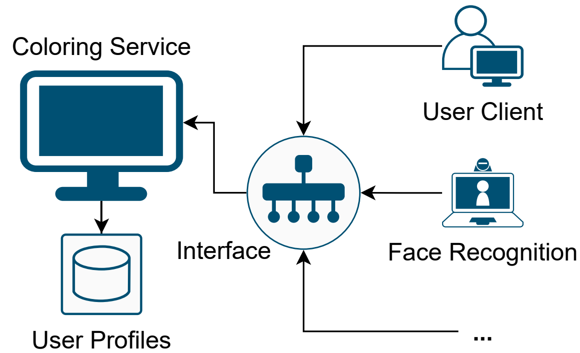


Figure 1. MAD-Coloring modular architecture concept.

The communication between clients and coloring should be kept as simple as possible to fit into the modular approach. This should enable easy access to the Coloring Service for new clients as well as the replacement of the Coloring implementation itself, e.g., if required by a change of the OS. To achieve this a simple interface should be created, supplying all required functionality for the coloring and client programs. New implementations could then simply use this interface and would be able to be used with the old clients/Coloring. This further enables the exchange of the communication framework with changes to only the interface itself. Clients and coloring would not be affected and could be used without any changes. The underlying communication framework should be chosen OS specific and use existing infrastructure rather than implementing something new. Therefore, reducing the implementation effort, requirements and overhead caused by MAD-Coloring, compared to an out-of-the-box solution shipped with a communication framework.

Due to the interface, client programs and Coloring Services can be created with minimal restrictions. For the prototype two client applications are planned. A manual user client, allowing the selection of moods by the user via text input, and a face recognition background service, detecting the mood via a webcam and changing the display coloring accordingly. Further, for the current prototype two Coloring Services are planned. Additionally to the already existing Hudkit-Coloring-

Service from our previous work a Gamma-Shift-Coloring-Service is planned, utilizing a vector based gamma shift. Following our modular architecture approach, the configuration files/user profile from one service can be used in the other service.

Thanks to the highly adjustable user profiles, changes or fine tuning of the color mapping in the implementation are always possible with little to no programming experience. This allows interested psychologists, therapists, and doctors to use the MAD-Coloring on their own. Enabling them to adjust the system according to their knowledge and research in the area of color psychology, highly tailored to their target group needs. Further, users trying the prototype on their own can adjust their profile as they most see fit, according to their preferences.

IV. MAD-Coloring

This section gives insight in the implemented MAD-Coloring prototype for Linux systems, including the different clients and Coloring Services. While some parts of it are Linux specific, like the specific D-Bus interface or gamma handling via Xrandr, the solution can easily be ported to Windows or macOS systems with small changes to the specific modules.

A. Interface

The interface is built on top of the D-Bus, as it represents the default solution for inter-process communication of most Linux desktop environments and therefore already is available on the system. This removes the need to install new software frameworks, perfectly fitting into the low overhead architecture of the concept.

The interface itself is separated into two parts. The main (top level) interface, providing simple python functions and a bottom level interface consisting of a D-Bus service and a corresponding client.

To ease the use of the interface and allow easy interchangeability of the underlying system the top level interface has been implemented only exposing the core functionality to client developers. It can be considered as a wrapper around the bottom level interface, providing the two required functionalities to implement new clients or a Coloring Service. These are a getter and setter functions for communication with the clients.

While the top level interface is rather simple, the bottom-line D-Bus communication is more extensive. The D-Bus service specified by it exposes an interface with three methods on the D-Bus session bus. However, only one of the three methods is currently used in the top level interface, a function to set the mood. The other methods provide interfaces for the currently active color code and the possibility to change the user profile on the fly, providing a boilerplate for more extensive clients. For the top level interface setter function, a connection to the exposed D-Bus interface is opened and the provided bottom level interface setter method is used to send the color change via the D-Bus.

It is up to the client if the communication is done via interface provided in this work or directly via the D-Bus. On the one hand, the baseline D-Bus service provides a richer API and might be used with any programming language or due to some universal command line interface tools. However, on the other hand, using the Python interface instead is less complex and allows an easy exchange of the underlying D-Bus framework, which might be required due to the change of the OS. Requiring only the getter and setter pair at the top level interface allows an easy exchange of the base interface, as the new framework must only realize these two functions. When sticking to Linux Desktops the D-Bus provides more extensive functionality and grants the exchange of the programming language, but the provided python interface grants interchangeability of OS and base interface. Choosing the right interface is up to the developer and should be decided individually according to the needs of the implementation.

B. User Profiles

The profiles are implemented as .conf configuration files. Using this well known and easy to use standard allows the modification of these files without any knowledge of programming. This is important, as for now, the prototype provides no GUI to edit the user profiles and the users have to edit their profiles with a text editor on their own.

In the scope of the prototype, the emotions and their respective coloring in the default profile are defined according the survey in Section II, which is shown in Listing 1. If the users want to change the color of an emotion they can either change the default profile or create a new user profile. Following an override approach, the Coloring Service will always check the specific user profile first. If the emotion is not defined in this profile, the service will fall back to the default profile. This makes it quite comfortable to specify some custom mood/color pairs while keeping default settings for most.

Listing 1. Example color definition

```
[Colours]
tired = #ede35a33
stressed = #5aeded20
sad = #35e85620
angry = #49d2fc20
unconcentrated = #a4f6fc20
anxious = #e8a63533
frustrated = #5aeded33
bored = #e8413555
```

It is also possible to define new emotion/color pairs in the user profiles, which are not defined in the default profile. However, to be used by the clients, the emotions have to be added to their codebase as well.

With the Gamma-Shift-Coloring-Service the user profiles were extended to also handle gamma shift values rather than only RGB values. This is detailed further in the Gamma-Shift-Coloring-Service section.

C. Hudkit-Coloring-Service

In the first version of our Coloring Service a combination between a modified version of Hudkit, introduced in Section

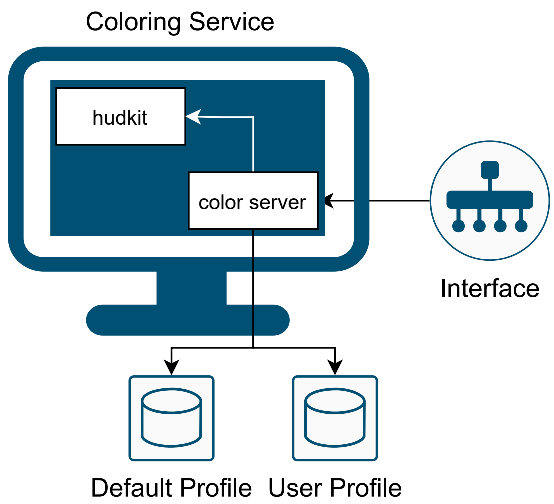


Figure 2. MAD-Coloring color service.

II, and a Python Color server is being used (Fig. 2).

Hudkit completely handles the display coloring. This happens by blending an RGBA colored image over the screen to taint it in the required shade. The alpha(α) blend of the display overlay is done via a linear calculation for every pixel on the screen. We define the tainting colors in the user specific color profile in hexadecimal encoded RGBA values. The color, which is displayed on the monitor is calculated using the "normal" color and the blend of the overlay, cf. Equation (2).

$$\vec{C}_{RGB} = \begin{pmatrix} Pixel_{red} \\ Pixel_{green} \\ Pixel_{blue} \end{pmatrix} * (1-\alpha) + \begin{pmatrix} Taint_{red} \\ Taint_{green} \\ Taint_{blue} \end{pmatrix} * \alpha \tag{2}$$

The color server provides the color services interface and control unit. This is achieved by implementing the earlier introduced Python interface and accessing the user profiles. Utilizing the Python interface, the server listens to changes of the mood send by a client. If a mood change is detected the server resolves the color according to the used profile. First, the user specific configuration is read, if no entry for the specific mood is found the default configuration is read.

After resolving the color, the server sends a signal with the determined color to the Hudkit web server. Should a mood not be defined in the configuration files, the color server defaults to black with low saturation and displays an error. With this error handling a missing emotion does not lead to further problems during the runtime, but clearly signals the user that a problem with the used client has occurred. This error color will not be interpreted as an emotion by the user, as black is not used by the default configuration and we highly discourage the usage of black as it is mostly related to negative emotions (cf. Table I). To counter possible negative emotions associated with black, the low saturation and higher brightness creates a grey overlay in the Hudkit server. This can be interpreted by users as boring, but will not trigger negative emotions.

D. Gamma-Shift-Coloring-Service

In this paper, we introduce a second kind of Coloring Service for our prototype where the display overlay Hudkit is exchanged against color shifting implementation utilizing the X.Org gamma correction via Xrandr, introduced in Section II. By doing so, we omit the display overlay software and reduce the performance overhead of the prototype.

The X.Org display service is handling the configured gamma correction. In our scenario, this gamma is configured via the Xrandr API. It is called from within the Python Coloring Services color server, after calculating the gamma correction for the configured color shift. We chose to use the binary Xrandr for the prototype rather than LibXOrg itself or its corresponding Python bindings, as the refined API provided by Xrandr greatly reduces the implementation complexity.

In order to follow the modular exchangeable architecture of MAD-Coloring and keep the same configuration files, an conversion from RGBA values to a gamma adjustment is required when using gamma correction instead of alpha blending. The major difficulty here is, that alpha blending is perceived linearly, while gamma correction is perceived exponentially by the human eye. Further, while alpha blending is displayed as an overlay, independent of the colors behind it, the gamma correction is highly dependent of its actual background. This means, that it is not possible to compute an overall gamma, which has the same effect for every background color as the corresponding alpha blend.

However, it is possible to calculate a gamma correction, which has the same effect as a RGBA alpha blending on a fixed background color via a RGB vector shift. The general formula for a vector based gamma shift on a single pixel can be seen in (3). The derivation for this equation can be found in Equation (8) (Appendix).

$$\vec{\gamma}_{RGB} = \begin{pmatrix} \frac{1}{log_{Pixel_R}(Pixel_R + (Taint_R - Pixel_R) * \alpha)} \\ \frac{1}{log_{Pixel_G}(Pixel_G + (Taint_G - Pixel_G) * \alpha)} \\ \frac{1}{log_{Pixel_B}(Pixel_B + (Taint_B - Pixel_B) * \alpha)} \end{pmatrix} \tag{3}$$

Calculating the correct gamma value for each pixel and coloring it accordingly would be absurdly resource intensive, anyhow it is possible to compute a general approximate γ value by using a fixed color vector. Determining the optimal base color vector for this use-case still needs further work, however, using the neutral grey (50% grey) color vector as displayed in Equation (4) provides an interim solution with adequate results.

$$\vec{C}_{50\%-grey} = \begin{pmatrix} 128 \\ 128 \\ 128 \end{pmatrix} \tag{4}$$

While using a fixed color vector as a base is no perfect solution, it is the most resource saving and yet functional approach. Resolving Equation (3) with Equation (4) we receive the formula used for the calculation of our vector based gamma shift, introduced in Section III, which can be seen in Equation (5)

$$\vec{\gamma}_{RGB} = \begin{pmatrix} \frac{1}{log_{128}(128+(Taint_R-128)*\alpha)} \\ \frac{1}{log_{128}(128+(Taint_G-128)*\alpha)} \\ \frac{1}{log_{128}(128+(Taint_B-128)*\alpha)} \end{pmatrix} \quad (5)$$

When looking up a color, the Hudkit-Coloring-Service transmits the color via the color server directly from the user profile to the display overlay. The Gama-Shift-Coloring-Service newly implemented in this paper first has to calculate the corresponding gamma shift value using the formula defined above.

In X.Org based systems the gamma correction is by default set to 1.0. For the Gamma-Shift-Coloring-Service we assume, that no changes were made to this default gamma correction by the user via system settings. However, we extended the color server and user profile with an option to define a default gamma correction value (red/blue/green-pre) which will then be taken into account for the calculation of the vector based gamma shift. Under the premise that the gamma shift from the Coloring Service is applied before any gamma correction defined in the system settings, the resulting screen color may be computed via the formula from Equation (6) representing the final formula of the Gamma-Shift-Coloring-Service. This formula is derived by inserting the resulting vector based gamma shift from Equation (5) as the pixels color value in the color vector formula for gamma from Equation (1). For better readability, a vector free display form was chosen.

$$\begin{aligned} \gamma_{red} &= \frac{1}{log_{128}((128+(Taint_{red}-128)*\alpha)^{\frac{1}{\gamma_{red-pre}}})} \\ \gamma_{green} &= \frac{1}{log_{128}((128+(Taint_{green}-128)*\alpha)^{\frac{1}{\gamma_{green-pre}}})} \\ \gamma_{blue} &= \frac{1}{log_{128}((128+(Taint_{blue}-128)*\alpha)^{\frac{1}{\gamma_{blue-pre}}})} \end{aligned} \quad (6)$$

In the context of the Gamma-Shift-Coloring-Service it might be confusing to set RGBA values in the user profile configuration file since they are not applied directly as with the alpha blend method from the Hudkit-Coloring-Service. We, therefore, added the option to define the emotion-color pair in the user profile with a gamma shift instead of a RGBA value. To extend the new usability feature with full compatibility to our old user profiles, we enhance the color server with a detection mechanism. The algorithm recognizes if an entry in the user profile configuration file contains an RGBA value or a gamma correction shift and accordingly handles the value. The Hudkit-Coloring-Service, however, still requires RGBA values and does not work with gamma shift values. Further extensions to the user profiles and coloring servers are planned, to overcome the current limitations.

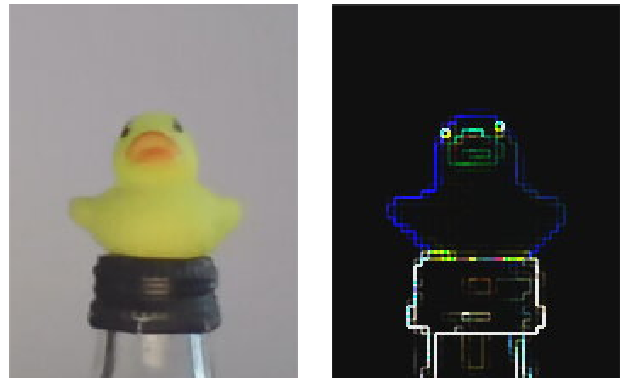


Figure 3. Face detection symbol image - feature highlighting.

E. Clients

The prototype includes two clients. A simple user client and an intelligent emotion detection client. The simple client provides a basic GUI to enter the current emotion and change the display coloring accordingly. It is implemented in Python and directly accesses the D-Bus instead of the Python wrapper interface.

The second client is an emotion recognition client, detecting the current emotional state of the user via image processing of a webcam feed. As a foundation the work of Rovai was used, creating a face recognition system utilizing a webcam with Python and OpenCV [11].

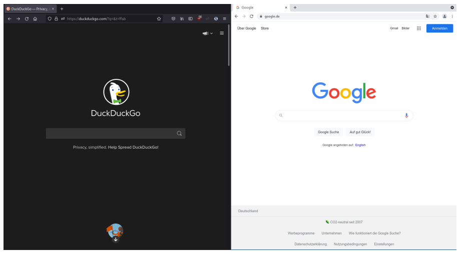
However, the final implementation differs fundamentally, as the client was extended to enable emotion recognition and connected to the Python interface to communicate with the Coloring Service. Further, a multi face detection was implemented, preventing a flickering color change if two or more faces are detected. As a result the client will not send emotional changes to the Coloring Service, until only one face is left in its field of view. A simplified version of the used feature detection can be found in Fig. 3. The eyes are clearly detected in white and the mouth in green. This allows the usage of the features in the neural network to determine the mood.

Due to the clean interfaces, between client and service, both clients are fully compatible with both Coloring Services versions. Therefore, no changes to the clients were necessary.

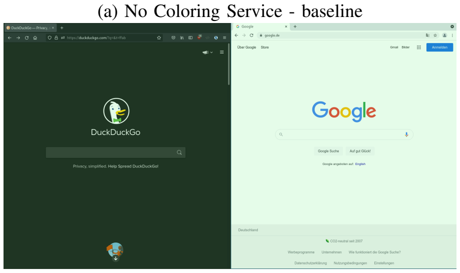
V. Comparison of first and second prototype

This section provides a quick overview between our current prototype and the version introduced in our previous work [1]. The focus is therefore centered around the new Gamma-Shift-Coloring-Service and the required introduced changes.

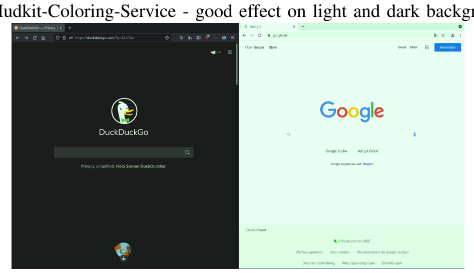
When comparing the first version of the prototype, which provided a display overlay utilizing alpha blending (Hudkit-Coloring-Service), with the second version, major graphical differences are visible. Since gamma correction is an exponential scaling algorithm the impact on the result depends strongly on the underlying colors. Figure 4 highlights the



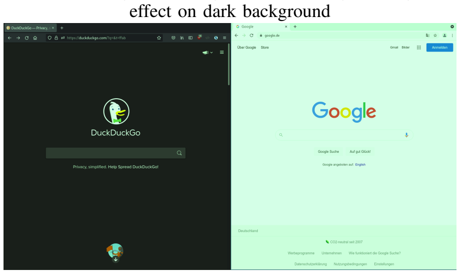
(a) No Coloring Service - baseline



(b) Hudkit-Coloring-Service - good effect on light and dark background



(c) Gamma-Shift-Coloring-Service - good effect on light background, minimal effect on dark background



(d) Gamma-Shift-Coloring-Service double opacity - extreme effect on light background, small effect on dark background

Figure 4. Comparison of different MAD-Coloring Color-Service effects for sad mood

comparison between applying alpha blending and using a gamma correction shift on dark and bright background images. In the images, it becomes clear, that the alpha blend takes much more effective to black areas than the vector based gamma shift approach does.

Considering the implementation aspects of both prototype versions, the technical differences are limited to the Coloring Service, which defines how the color adjustment is displayed and the configuration reader, as well as the connected user profiles. On the one hand, the configuration files and their parsing has become slightly more complex in the second version of the prototype, to enable the configuration and usage of gamma values. On the other hand, the inter-process communication procedure of the displaying interface as well as the overhead of the Coloring Service has been greatly reduced, making the whole MAD-Coloring application more lightweight. As a downside the calculations for the vector based gamma shift are significantly more complex, however, this only concerns the implementation of the prototype and has no negative effect on the end user.

Taking the technical and visual changes into account the vector based gamma shift provides a solid option for computers with lower tech-specs, allowing a more resource-conserving execution of MAD-Coloring. To counteract the weaker coloring effects of the gamma shift on dark backgrounds, compared to the alpha blend method, the usage should be limited to applications in light-mode, e.g., Excel, Office or browser usage. Summarizing, the vector based gamma shift Coloring Service provides a great solution for weak office machines, while the alpha blend Coloring service is the all-rounder with higher overhead, e.g., for gaming use-cases.

VI. Case Study

Our current team solely consists of researchers from the IT domain, we have not yet consulted any psychological/medical experts. Thus, we conducted a technical case study instead of a psychological evaluation of our prototype. The study was conducted on a Lenovo ThinkPad with 14GB RAM, an AMD onboard graphic chip, using a GNU/Linux operating system with a gnome X.Org desktop. During the test runs checks regarding usability impairments were conducted.

To pre-empt ethical concerns without the approval of an ethic-committee our test user simply controlled MAD-Coloring and rated possible concerns regarding the readability and usability of the desktop with activated MAD-Coloring. However, he was not exposed to the software for longer periods. The face recognition was triggered with prepared photos instead of live images of the test user.

As a scenario, a computer science student with no prior experience of MAD-Coloring was instructed to start the MAD coloring client and conduct a manual color change via the simple user client. Afterward the face recognition client had to be started. He was supplied with a computer that had a preinstalled MAD-Coloring Service and the MAD-service's Linux manual page.

The user was able to activate the service, change the color and start the face recognition client in less than a minute. All color changes and emotion detections worked without any problems. The user did not experience any problems regarding reduced readability, yet the usability in color sensitive applications like image editing was highly impaired because of MAD-Coloring's color overlay. However, this was an expected side effect as mentioned in our concept (cf. Section III). The effect of MAD-Coloring for some emotions can be seen in Fig. 5.

Following, we suggest a method for measuring the effects of MAD-Coloring in coming evaluations, which still needs to be reviewed by psychological experts. Since MAD-Coloring is designed to be used in daily routines and any kind of test scenario might cause discrepancies, we strongly recommend to evaluate MAD-Coloring integrated into the daily life of subjects.

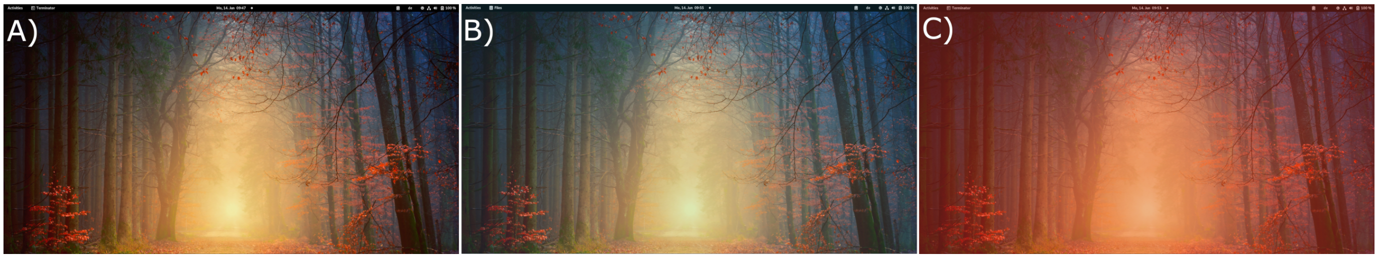


Figure 5. MAD-Coloring in effect for moods A) neutral B) mad C) bored.

To quantify MAD-Coloring, we introduce the concept of a mood diary where subjects record their emotional condition, and whether they are able to concentrate for work or not. The subjects might write this mood diary for a reasonable period (2-4 weeks). Afterwards, based on these diaries, basic emotional profiles can be created for all subjects.

In the second phase we recommend to split up the subjects into three groups. Group A will be working with our MAD-Coloring and the pre-configured color profile (based on scientific work from the color psychology domain and our own survey). Group B is working with MAD-Coloring as well, but with "anti"-colors, which have been associated to be negative according to a specific mood (e.g., red if the subject is already mad). Group C is the control group, which continues working without any influence from MAD-Coloring. This phase should be conducted over a larger time period (2-4 months or longer), as most likely some time is required to get used to the color changes. Especially at the beginning these changes could have a negative impact on the subjects.

By comparing the deltas of the three groups the essential effects of MAD-Coloring can be determined and whether the effect depends on specific colors or just generally on shifting these. Afterwards further evaluations can be planned targeted on the existing data.

VII. Related Research

While the idea of color psychology is not new, there is to our best knowledge no closely related research in the HCI context. However, some other research topics in the context can be viewed as relevant.

A. Blue Light Filtering

At the first sight blue light filtering software seems to differ quite significantly from our solution, but the fundamental ideas are quite similar. Both software solutions modify the color shade of the display to obtain effects on the human user. However, blue light filtering is based on different medical effects (cf. [23]) and in most cases it is implemented completely different than MAD-Coloring.

While MAD-Coloring is based on the psychological effect of colors and the emotions triggered by colors, the idea of blue light filtering, as described in [24] is based on physical and bio-chemical effects [25]. They determine that blue light emitted by screens contains more energy than any other color. Further, it is more exhaustive for the human eye than other colors. In addition to the physical aspects blue light suppresses the production of the sleeping hormone melatonin which can cause sleeplessness.

As aforementioned, blue filtering software is often implemented completely different than our solution. Its common among blue filters to adjust the alpha channel to suppress parts of the blue light. The resource costs for this approach can be expected to be less than those for overlay based filters. This is due by the fact, that the graphics card is not required to compute a translucent overlay in. However, for our MAD-Coloring system an alpha shift approach does not fit the requirements since the software needs to display tints in various colors.

B. Colors and Trust in User Interface Design

Hawlitschek et al. [2] thematize the influence of colors on trustworthiness of user interfaces. They analyzed the moods and meanings associated to different colors via an experiment.

In this experiment, the probands had to pass a finance based trust experiment. They were provided with GUIs in different hues and small amount of money was handed to each proband. Then, by transferring money to other probands, they where able to increase the value of their sum by trading between each other. The experiment tried to determine if the color of their GUI had an impact on the trust they have in the other probands. However, they where not able to gain meaningful results from this experiment.

C. Effect of Colors on Emotions in Games

Joosten et al. [26] empirically studied the emotional effect of color on players of a video game. As a foundation they used a subset of the color⇒emotion arrangements of Plutchik [27], allocating the emotional effect of colors to an emotion as following; dark green ⇒ fear, light blue ⇒ surprise, red ⇒ anger, and yellow ⇒ joy.

The evaluation was conducted on 60 participants which all had to play five levels of a video game programmed for this evaluation. The first level acted as a control level, being the same for all players, while the following four levels had 24 possible deviating color arrangements. These were evenly distributed on the players, leaving 2-3 players per arrangement.

With this setup the expected negative arousing effect of red, as well as the positiv effect of yellow could be confirmed, while the other two colors did not had a meaningful effect on the players emotional state.

VIII. Conclusion

In this paper we presented the second version of our novel MAD-Coloring framework, highlighting a Hudkit-free Coloring Service. Respecting the basics of color psychology a second, more lightweight, Coloring Service prototype was implemented, providing a transparent, color adjustable overlay for Linux X-desktop systems utilizing a vector based gamma shift approach. This new service was tested with our two earlier clients. A simple input client, allowing the manual change of the display color and an emotion recognition client, detecting the users current emotional state via a webcam and adjusting the display color accordingly. We also extended our user profiles with options to directly enter gamma values, to increase the ease of use with the new Coloring Service.

MAD-Coloring, in combination with this clients, is capable to display a decent, transparent overlay over multiple desktops according to the users current emotion. The new gamma based version of the Coloring Service provides similar results as the initial alpha blend version. However, the coloring over dark backgrounds is harder to notice, potentially reducing the effect. Then again, the technical footprint of MAD-Coloring was greatly reduced with the new gamma shift approach, removing the overhead introduced by Hudkit.

While this version of the Coloring Service provided overall slightly worse coloring effects it can provide a great alternative for low spec computers, especially for office work with bright backgrounds and dark fonts, e.g., Excel or Word tasks.

While fully functional, further work is required to refine and improve the system. On the one hand, medical studies are required to evaluate the psychological impact of the system and therefore confirming its usefulness. Following this further studies are required to find optimal color profiles to maximize the effect. On the other hand, further technical improvements can be conducted. The support of more desktop environments could be realized and more clients should be implemented, allowing more specific use cases and optimal support for more kinds if needs. These clients could also use smart devices like watches or fitness tracers, allowing the integration of blood pressure into the emotion recognition.

Acknowledgments

We want to express our gratitude towards Andrea Steiner for her contribution towards the displayed mathematical content of this paper, providing guidance and corrections.

Further we want to thank Jakob Loskan for his advice concerning the technical details behind alpha blend and gamma shift, as well as the mathematics involved.

References

[1] L. Brodschelm, F. Gräber, D. Hieber, and M. Hermann, "Mood adaptive display coloring - utilizing modern machine learning techniques and intelligent coloring to influence the mood of pc users," in *ACHI 2021, The Fourteenth International Conference on Advances in Computer-Human Interactions*, 07 2021, pp. 48–54.

[2] F. Hawlitschek, L.-E. Jansen, E. Lux, T. Teubner, and C. Weinhardt, "Colors and trust: The influence of user interface design on trust and reciprocity," in *2016 49th Hawaii International Conference on System Sciences (HICSS)*. IEEE, jan 2016, pp. 590–599.

[3] R. M. Rider, "Color psychology and graphic design applications." Liberty University, 2010.

[4] M. Kido, "Bio-psychological effects of color," *Journal of International Society of Life Information Science*, vol. 18, pp. 254–268, 2000.

[5] M. A. Maier, A. J. Elliot, and R. A. Barton, "Color in achievement contexts in humans," in *Handbook of Color Psychology*, A. J. Elliot, M. D. Fairchild, and A. Franklin, Eds. Cambridge University Press, pp. 568–584.

[6] R. A. Ainsworth, L. Simpson, and D. Cassell, "Effects of three colors in an office interior on mood and performance," *Perceptual and Motor Skills*, vol. 76, no. 1, pp. 235–241, Feb. 1993.

[7] I. Kondratova and I. Goldfarb, "Color your website: Use of colors on the web," in *Usability and Internationalization. Global and Local User Interfaces*, N. Aykin, Ed. Berlin, Heidelberg: Springer Berlin Heidelberg, 2007, pp. 123–132.

[8] A. J. Elliot and M. A. Maier, "Color-in-context theory," in *Advances in Experimental Social Psychology*. Elsevier, 2012, pp. 61–125.

[9] C. Li and H. Shi, "Medical space oriented color psychology perception model," *Applied Mechanics and Materials*, vol. 587-589, pp. 461–467, 07 2014.

[10] K. R. Alexander and M. S. Shansky, "Influence of hue, value, and chroma on the perceived heaviness of colors," *Perception & Psychophysics*, vol. 19, no. 1, pp. 72–74, Jan. 1976.

[11] M. Rovai, "Real-time face recognition: An end-to-end project," 2018, last visited: 2021.06.14. [Online]. Available: https://towardsdatascience.com/real-time-face-recognition-an-end-to-end-project-b738bb0f7348

[12] D. Yang, A. Alsadoon, P. Prasad, A. Singh, and A. Elchouemi, "An emotion recognition model based on facial recognition in virtual learning environment," *Procedia Computer Science*, vol. 125, pp. 2 – 10, 2018, the 6th International Conference on Smart Computing and Communications.

[13] S. Ghosh, E. Laksana, L.-P. Morency, and S. Scherer, "Representation Learning for Speech Emotion Recognition," *Interspeech 2016*, pp. 3603–3607, Sep. 2016.

[14] A. R. Smith, "Image compositing fundamentals," Tech. Rep., 1995.

[15] P. Berndt and other, "Pqiv git repository," 2021, last visited: 2022.05.24. [Online]. Available: https://github.com/phillipberndt/pqiv/

[16] L. C. Klopfenstein and other, "Ontopreplica git repository," 2021, last visited: 2022.05.24. [Online]. Available: https://github.com/LorenzCK/OnTopReplica

[17] (2021) The hudkit-projekt github page. Last visited: 2021.06.14. [Online]. Available: https://github.com/anko/hudkit

[18] J. de Vries, *Learn OpenGL - Learn Modern OpenGL Graphics Programming in a Step-by-step Fashion.* Kendall & Welling, 2020.

[19] J. D. Foley, A. van Dam, S. K. Feiner, J. F. Hughes, and R. L. Phillips, *Introduction to Computer Graphics*, 1st ed. Addison-Wesley, 1994.

[20] "Microsoft documentation - using gamma correction," 2021, last visited: 2021.10.25. [Online]. Available: https://docs.microsoft.com/en-us/windows/win32/direct3ddxgi/using-gamma-correction

[21] "Xorg - lib documentation," 2021, last visited: 2021.10.25. [Online]. Available: https://x.org/releases/current/doc/libX11/libX11/libX11.html#Introduction_to_Xlib

[22] "Freedesktop - xrandr," 2021, last visited: 2021.10.25. [Online]. Available: https://xorg.freedesktop.org/archive/current/doc/man/man1/xrandr.1.xhtml

[23] (2020, jul) Harvard health letter - blue light has a dark side. Last visited: 2021.06.14. [Online]. Available: https://www.health.harvard.edu/staying-healthy/blue-light-has-a-dark-side

[24] S. Mitropoulos, V. Tsiantos, A. Americanos, I. Sianoudis, and A. Skouroliakou, "Blue light reducing software applications for mobile phone screens: measurement of spectral characteristics and biological parameters." in *RAP 2019 Conference Proceedings*. Sievert Association, 2020, pp. 220–224.

[25] R. Sutherland, "What is in a color? the unique human health effects of blue light," in *Envirionmental Healt Perspectives volume 118*. News Focus, 2010, pp. 23–27.

[26] E. Joosten, G. Lankveld, and P. Spronck, "Colors and emotions in video games," *11th International Conference on Intelligent Games and Simulation, GAME-ON 2010*, 01 2010.

[27] R. Plutchik, "The nature of emotions: Human emotions have deep evolutionary roots, a fact that may explain their complexity and provide tools for clinical practice," *American Scientist*, vol. 89, no. 4, pp. 344–350, 2001. [Online]. Available: http://www.jstor.org/stable/27857503

APPENDIX

SINGLE CHANNEL SHIFT GAMMA EQUATION

The color/γ-vector is only used for the ordered display of the color, while mathematically only the skalar product is used. Therefor, the vectors can be separated into its components and be compared pairwise. Equation (7) shows this for the red channel from Equations (1) and (2).
In the following Pixel is abbreviated with P and Taint with T.

$$\begin{aligned} & {P_R}^{\frac{1}{\gamma_R}} = P_R * (1-\alpha) + T_R * \alpha \\ \Leftrightarrow\quad & {P_R}^{\frac{1}{\gamma_R}} = P_R - P_R\alpha + T_R * \alpha \\ \Leftrightarrow\quad & {P_R}^{\frac{1}{\gamma_R}} = P_R + (T_R - P_R) * \alpha \\ \Leftrightarrow\quad & \frac{1}{\gamma_R} = log_{P_R}(P_R + (T_R - P_R) * \alpha) \\ \Leftrightarrow\quad & \gamma_R = \frac{1}{log_{P_R}(P_R + (T_R - P_R) * \alpha)} \end{aligned} \tag{7}$$

GAMMA SHIFT VECTOR EQUATION

This equation provides the basis for the vector based gamma shift. It can be derived by equating Equations (1) and (2).
In the following Pixel is abbreviated with P and Taint with T.

$$\begin{aligned} & \begin{pmatrix} {P_R}^{\frac{1}{\gamma_R}} \\ {P_G}^{\frac{1}{\gamma_G}} \\ {P_B}^{\frac{1}{\gamma_B}} \end{pmatrix} = \begin{pmatrix} P_R \\ P_G \\ P_B \end{pmatrix} * (1-\alpha) + \begin{pmatrix} T_R \\ T_G \\ T_B \end{pmatrix} * \alpha \\ \Rightarrow\quad & Applying\ Equation\ (7)\ for\ all\ 3\ channels \\ \Rightarrow\quad & \begin{pmatrix} \gamma_R \\ \gamma_G \\ \gamma_B \end{pmatrix} = \begin{pmatrix} \frac{1}{log_{P_R}(P_R+(T_R-P_R)*\alpha)} \\ \frac{1}{log_{P_G}(P_G+(T_G-P_G)*\alpha)} \\ \frac{1}{log_{P_B}(P_B+(T_B-P_B)*\alpha)} \end{pmatrix} \end{aligned} \tag{8}$$