A digital framework for color design and education

[1]Pietro Paglierani
pietro.paglierani@gmail.com
[corresponding author]
[2]Francesca Valan
francesca@francescavalan.it

[1]Independent Consultant
[2]Francesca Valan Studio

ABSTRACT

Color design is assuming a role of greater and greater importance in industry and architecture. However, only a few tools can effectively support designers and architects in their color-related activities, besides color atlases or color measurement devices. On the other hand, many significant results are now at hand in the field of computational color. In particular, various software libraries are available in numerical simulation environments or popular programming languages such as Python. Unfortunately, architects and color designers traditionally do not have a background in signal processing and/or programming techniques. Consequently, the use of such tools is limited to a relatively small community of color scientists, while the much broader spectrum of color-related activities in the industrial world or in architecture cannot benefit of such a collection of color science algorithms and transformations. To fill this gap, this paper presents a set of digital functions, named *"ColorTools"*, collected in a single framework written in Python. ColorTools makes use of (and eventually extends) the most popular color-theory libraries now available, and presents to the user a high level and friendly interface, specifically designed to match the needs of color designers. To demonstrate the effectiveness of the proposed framework, the paper describes the use of ColorTools in some real color projects. Finally, ColorTools can also be used in color education, since it allows exploring many basic color theory concepts.

Pietro Paglierani received the Master degree and the Ph.D. degree in Electronic Engineering from the University of Padua. He is currently a project leader in the R&D Department, Italtel S.p.A. His main interests are in digital signal processing, image and video processing, Network Function Virtualization, and Edge Computing. He has actively contributed to the definition of ITU and ETSI standards related to network performance assessment and published more than fifty papers on this topic.
He has participated in many national and European (FP7 and Horizon 2020) funded research projects in the Information and Communications Technology sector. He was previously with Tektronix Padova, Necsy S.p.A. and Consorzio Padova Ricerche, where he was involved in the design of signal processing applications. Since 2012, he is a certified Scrum Master.
Francesca Valan is an industrial designer, specialized in the design of colors, materials, and finishes (CMF Design). She graduated in Industrial Design at the IED in Milan in 1989 and received her Master in Surface Quality in 1990. She lives and works in Milan, where she founded her studio in 1998. As an Industrial Designer, her activity consists in defining Product Identity. Her CMF projects range from elevators to office furniture, from home appliances to sport items, pens, glasses etc. She has collaborated with De Longhi Group, Lechler, Samsung, Telecom Italia, LG, Hitachi, Olivetti, Atomic, Technogym, Fila, Campari, and many others. She is a teacher in university and master courses in color design technology. She developed a method of prediction of trends based on historical and statistical analysis of color by market sector, which exploits the periodicity of color preferences and the duration of cycles for product type.

1. INTRODUCTION

In the complex and extremely competitive global market of these days, Color Design has assumed a role of great importance for industry. At the same time, new social trends and customer behaviors are profoundly changing the way designers choose colors. For instance, with the advent of online commerce, customers very often select products from the screen of a personal computer, of a tablet or even of a smartphone, i.e. through means that can severely distort the actual chromatic appearance of an object.

Consequently, the role of designers is rapidly changing, since they must not only select trendy colors, to give products an appealing and specific character. Instead, they are more and more involved in all those processes of the industrial production chain on which color selection can have an impact.

In such activities, designers must deal with many aspects that require an in-depth knowledge of color theory such as, for instance, color coding (for clear and precise data communication to providers and production units), color quality (to guarantee color reproducibility, color constancy, etc.), or all the possible attributes that finishing can assume (solid, metallic, glossy, matt, etc.)

However, very few specific *"color tools"* can support designers in these tasks. One of the most important is the Color Atlas, be it Munsell or NCS, i.e. a collection of coded physical samples usually printed on paper, whose closeness to a set of corresponding pre-defined colors is certified, with a given accuracy, by a specialized lab (Cochrane, 2014), (Landa and Fairchild, 2005), (NCS – We're all about colour, 2018).

Besides the Color Atlas, several commercial color measurement devices are now accessible at reasonable costs. Such devices can be of two types, i.e. colorimeters or spectrophotometers (Color Measurement, 2018), (Design Industry; X-Rite, 2018). The first aim at measuring color by reproducing the behavior of the human vision system, to provide perceptually meaningful data, while the latter perform a sample analysis of the color spectrum, thus obtaining results that can go beyond the vision capabilities of humans. Very often, color measurement devices are provided with a supporting proprietary software, which can be used for configuration and management purposes, but also offers color processing capabilities (Design Industry; X-Rite, 2018). Typically, such software can implement few essential functions, such as color conversion between different spaces (Lab, NCS, Munsell), or perceptual comparison of measured data.

On the other hand, many results are now available in the field of computational color, i.e. that branch of signal processing specifically focused on color science. In particular, many software libraries for color science are now available in well-known numerical simulation environments, such as Matlab or Scilab, or written in popular programming languages, such as Python (The Munsell and Kubelka-Munk Toolbox, 2010), (Colour Science for Python, 2018). For instance, (Colour Science for Python, 2018) is a Python software package that implements a vast number of color theory transformations and algorithms, which could be very useful in many color design activities.

Unfortunately, while such simulation environments and programming languages are well known to engineers, physicists and computer scientists, they result wholly unfamiliar to color designers, who traditionally do not have a background in signal processing and/or programming techniques. Consequently, the use of such tools is confined to a relatively small community of color scientists, and to their experimentations and research made in specialized labs. Conversely, the much broader spectrum of color-related activities carried out in the industrial world and in architecture cannot benefit from such a wide and well-established collection of software tools.

To fill this gap, this paper presents a set of software functions, named *"ColorTools"* for architects and designers, collected in a single framework written in the Python programming language. The proposed software makes use of (and eventually extends) the most popular color-theory libraries, and presents to the user a high level and friendly interface, specifically designed to match the needs of a color designer. For the sake of brevity, however, this paper will present and briefly discuss only a few real-life examples of its use in color projects.

In particular, we will first consider the problem of selecting a color set for a new line of sunglass frames, using the Munsell Color System (ASTM Standard D 1535-14, 2018). The developed tool allows digitally exploring and processing the colors in the Munsell color space straightforwardly, thus greatly enhancing and at the same time simplifying the color selection process. Also, the problem of color distortion due to the exposition of the products in a digital store accessible through the Internet is considered; the support given by ColorTools in this context is described and briefly discussed. Then, the problem of developing a graphical project taking into account color visual deficiency is introduced, and the use of ColorTools to analyze the effects of color vision deficiency in architecture (the Dino Compagni school in Florence) is presented.

Finally, we highlight that the ColorTools framework can also be used in color education activities, since it allows exploring, simply and

straightforwardly, many basic color theory concepts that play a crucial role in color science, from color space to visual deficiency, color adaptation, color harmony, and many others.

2. THE COLORTOOLS

Several specific software tools are available in the field of color science. The open source (The Munsell and Kubelka-Munk Toolbox, 2010) implements a set of routines that can be used either in MATLAB or Octave, related to the Munsell system and the Kubelka-Munk model, as well as to other areas of color science. Currently, however, though contributions in many programming languages can be found, most of the open source software for color science is written in Python. For instance, a broad set of color functions are included in (Colour Science for Python, 2018), which covers various color-science domains, from color models to optical phenomena, and includes part of the functions developed in (The Munsell and Kubelka-Munk Toolbox, 2010). (Python-Colormath, 2014) is another, simpler Python module, which can give support for various color spaces, offering conversion, chromatic adaptation and color distance calculation functionalities. Besides these specific modules, popular Python packages for signal and image processing (such as Numpy or OpenCv) can provide useful functions for color science-based activities.

However, to the best of the authors' knowledge, a common framework that combines all such functionalities into a single application specifically tailored to the needs of color designers is not available yet. To fill this gap, this paper introduces ColorTools, a recently developed software application now available in a prototypal version.

The use of ColorTools can give various benefits to color designers. Firstly, the color science functionalities included in the framework have been thoroughly tested and characterized. The combination of functions from different packages, each with its specific interface, can originate various types of errors, malfunctioning or limitations. A second motivation for the development of ColorTools is that even combining all the available Python packages, specific functionalities of interest for Color Designers are still not straightforwardly available; hence, ad hoc ColorTools modules (such as ColorMix or Dalton, which will be discussed in the following) had to be developed. Also, most of the functions offered by the available open source projects have limitations, and must be used within specific contexts. For instance, the Munsell interpolation tool in (Colour Science for Python, 2018), which can transform Munsell colors into CIE 1931 Yxy coordinates and vice-versa, can

be applied only when the Munsell Value is in the range [1-9], and the Munsell Chroma is greater than or equal to 2. For this reason, a specific tool had to be developed for neutral Munsell colors, based on (Munsell Renotation Data, 2015).

Finally, ColorTools presents a simple multi-instance graphical interface, which allows to straightforwardly using the provided color science functionalities, thus speeding up color design activities, and hiding all the problems related to the use of complex and unfamiliar simulation environments.

2.1 A BRIEF INTRODUCTION TO COLORTOOLS

The ColorTools framework offers a (prototypal) graphical interface through which the user can launch any number of concurrent functionalities related to Color Theory and Color Design. The initial ColorTools view presented to the user is shown in figure1.

From the initial window, the user can launch different functionalities, through the buttons in the window. In particular, the following functions are presently available, or under development:

- ColorMix: starting from two colors defined in the Munsell or Lab color spaces, a weighted mixed color can be obtained, and its Munsell and Lab code is provided.
- Dalton: shows how a person affected by Color Visual Deficiency would see a given image.
- Covert: conversion tool, between the most used color spaces (Munsell, Lab, Yxy, RGB, etc.).
- ColorPick: evaluates the average Munsell/Lab color in an area specified by the user through the mouse in a given image.
- MunsellBook: digital (non-reliable, with commercial PC screens) representation of Munsell Colors, for educational purposes.
- DarkMunsell: conversion between Munsell and Yxy color spaces, for low-value Munsell colors (value lower than 1).
- ColorDistance: color distance evaluation, using CIE76, CIE94 and CIEDE2000.
- Gamut: verifies if a color is within a specific gamut.
- Adaptation: implements color adaptation algorithms (Von Kries, Bradford)
- Version: shows the ColorTools Software Version.

As discussed in the following, the functions described above have been inserted in the

However, this simple formulation hides indeed an extremely complex problem. A huge number of studies have been dedicated to color harmony through the centuries up to nowadays, in art, design, psychology and color science by authors such as Newton, Goethe, Chevreul, von Helmholtz, Munsell, Itten, and Albers, to quote just a few (Burchett, 2002), (O'Connor, 2010).

However, despite all the efforts dedicated to this topic, consensus regarding color harmony still lacks, and the related information can often result contradictory or ambiguous (O'Connor, 2010).

ColorTools to respond to the practical needs and problems encountered in their activities by Color Designers.

ColorTools offers a graphical interface through which the user can launch any number of concurrent functionalities. The user can freely jump from one function window to any other, as shown in figure 2.

Such functions can be easily upgraded, modified, and new ones can be straightforwardly inserted. In the following, for the sake of brevity, only two of them, i.e. ColorMix and Dalton will be briefly described and discussed.

For this reason, when facing the practical problem of selecting colors for a commercial product, designers usually start from some of the rules available in the literature, chosen and organized according to a personal, proprietary *"recipe"*, which can depend on professional experiences, previous studies, cultural preferences, personal attitudes, etc. This way, the process of searching and selecting colors typically results in a heuristic procedure, in which an initial color set is iteratively modified and adjusted until a subjectively satisfactory result is achieved.

3 THE COLORMIX USE CASE

A basic activity in many color design projects is the definition of a set of colors for a product, which should result attractive to potential customers.

The problem of selecting colors that originate a positive aesthetic response usually relates to the theory of color harmony. A simple and generally accepted definition of harmonic colors is given in (Burchett, 2002):

Any discussion about color harmony, however, is out of the scope of this paper; the interested reader can find a survey, recent results and significant references in (O'Connor, 2010), (Burchett, 2002). Conversely, ColorTools can provide some useful and flexible functionalities that support designers in the color selection process.

A simple pragmatic approach to select colors follows the empiricist suggestions of Chevreul (which influenced many impressionist painters), Munsell and Itten, according to which harmony relates to complementary colors, i.e. colors that when combined produce a neutral color (O'Connor, 2010). On the other hand, although

"Colors seen together to produce pleasing affective response are said to be in harmony".

pairs of complementary colors (red-green, blue-yellow, white-black) play an essential role in the human vision system, there is no experimental evidence that the balance of complementary colors can automatically originate a positive aesthetic response. Nonetheless, the possibility to explore the color space, by merely modifying colors though ad hoc combinations can significantly help and speed up the color selection process.

In this context, one useful functionality made available by ColorTools is ColorMix, which provides the possibility of mixing two colors, starting from the Munsell or Lab color spaces. The mixing process considered here is performed in the CIE XYZ color space, through the weighted Euclidean vector sum of the colors expressed in XYZ coordinates. Conversely, this paper does not consider the problem of mixing paints, which is usually addressed by the Kubelka-Munk model, once the absorption coefficient, K, and scattering coefficient, S for each paint and at each wavelength are known. The Kubelka-Munk model and the related tools to estimate K and S, can be found as Matlab functions in (The Munsell and Kubelka-Munk Toolbox, 2010). Such tools are also available in Python in (Colour Science for Python, 2018). However, they have not been included in the ColorTools yet.

The digital color mixer, given two colors in the Munsell or Lab color space, can estimate the corresponding mixed color. The graphical interface of ColorTools allows to enter the original colors in the selected space, and a slider to change the weights of the originating colors. Also, the output color is expressed in the Munsell or Lab color space. Moreover, both the input and the output colors are visually represented. Unfortunately, the reproduction of input and mixed colors on the screen, in general, cannot be considered reliable. An accurate visual representation can be obtained only if the output color is within the used screen gamut, and the screen is calibrated (this point will be discussed later).

An example of the graphical interface for the ColorMix function is reported in figure 3.

To obtain the mixed color, ColorMix first converts the input colors from the original color space to the CIE1931 XYZ color space (The Munsell and Kubelka-Munk Toolbox, 2010), (Richter, 2012). Mixing is then performed by exploiting convexity and linearity of the XYZ space. In fact, for any two color C1 and C2 inside the human visual gamut, any linear combination of such colors still is within the gamut (Richter, 2012).

The mixed color is obtained as a linear combination of the two original colors performed in the XYZ color space, i.e. $C = (1-\alpha)C1 + \alpha C2$, and is finally transformed to the Munsell or Lab color space.

An example of the use of the ColorMix function to select colors is the Skin Tone Project. In this project, the objective was to create a set of colors for sunglass frames that could result in harmony with the colors of the skin of the final users. The complexions of 50 persons of different nationalities including all phototypes (from PHOTOTYPE I, Lactea / reddish to PHOTOTYPE VI, Very Dark / Black) were measured using the Munsell Color System by visual interpolation, as specified in (ASTM Standard D 1535-14, 2018).
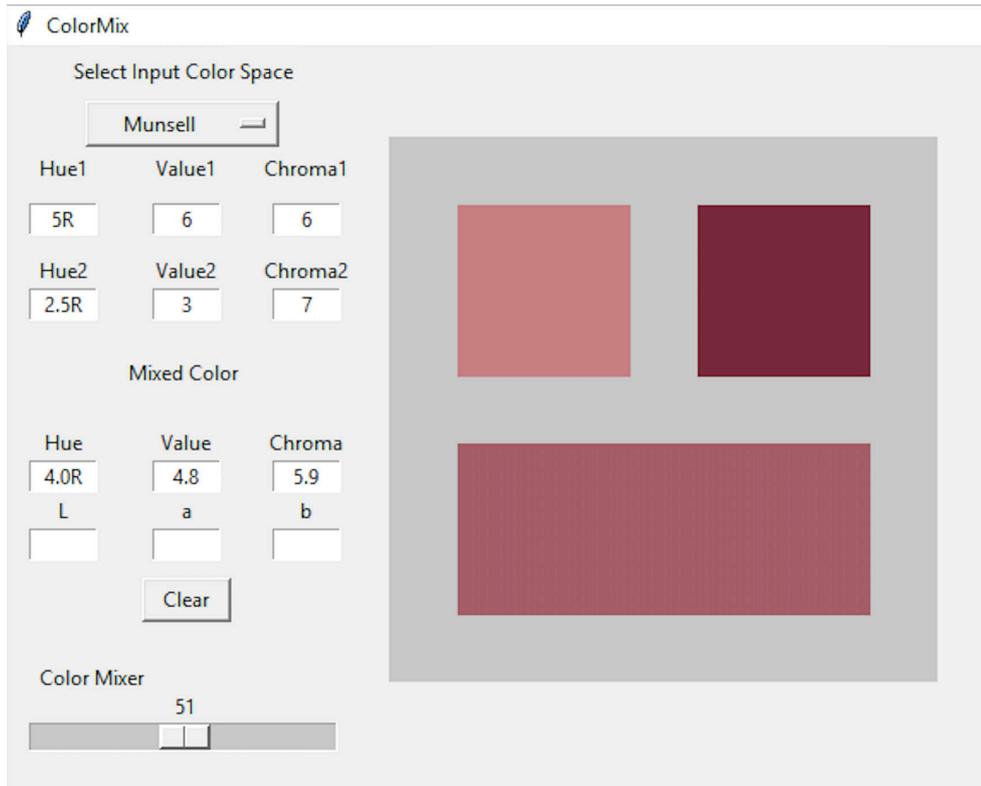
Figure 3 - ColorMix, two input colors in Munsell or Lab are mixed according to selectable weights.

Moreover, also the most difficult-to-match frame colors (vivid green, blue, vivid, orange, white) were sampled in the Munsell color space. With the COLORMIX function, interpolations were then obtained between skin and frame colors, in the search for skin-frame color harmony.

A detailed description of the Skin Tone project is out of the scope of this paper. However, using ColorTools, different approaches were quickly adopted in the search of candidate sets of colors. In the end, the selected set was obtained by suitably mixing average skin colors, with a previously selected set of sunglass frame colors. Different values of the combination coefficient α were tested, to determine the most satisfactory combinations, which was obtained by adding to the sunglass frame colors average skin colors multiplied by low values of the α coefficients.

Another critical aspect that can be analyzed with ColorTools, is the reproduction of colors when visualized on a digital screen. With the advent and rapid success of online commerce, customers very often select products through the Internet, using a personal computer, a tablet or a smartphone. However, digital screens can severely distort the actual chromatic appearance of objects, if the selected colors do not fall within the screen gamut (Rec. ITU-R BT 709-6, 2015), (Rec. ITU-R BT 2020-2, 2015). It should be noticed that, even in this case, color reproduction could be inaccurate, if the screen has not been calibrated. Nonetheless, selecting colors outside the gamut inevitably results in a distortion of the chromatic appearance of a product, independently of the quality of the screen used by the customers.

To verify the position of a color with respect to a given gamut, ColorTools offers the Gamut functionality. An example of its use is shown in fig.4, where a set of data, in this case originated by ColorMix, is positioned on the CIE xy chromatic plane, together with the reference gamut of a High Definition (HD) screen (Rec. ITU-R BT 709-6, 2015) and an Ultra High Definition (UHD) screen (Rec. ITU-R BT 2020-2, 2015). On the right, the plotted colors are shown, except when they fall outside the screen gamut.

As shown in fig.4, one can easily verify when selected colors are in or out of the gamut of interest.

4 THE DALTON USE CASE

In most graphical projects, the primary objective of color design is increasing readability of the relevant information, for instance, to highlight alert signals, or just to attract the attention of potential customers. However, if this goal is achieved for the majority of people with a normal visual system, it can be missed entirely for that part of the population that is affected by color visual deficiency. Since about 8% of the male population suffers from color-blindness, not taking into account this problem could lead to unsatisfactory results, or even to severe problems, in particular when some kind of vital information, such as alert signals, is of interest (Machado, Oliveira, and Leandro, 2009). Thus, simulating what color-blind can see is a fundamental activity in most graphical color design projects, as well as in architecture. The Dalton function of ColorTools allows simulating the vision of people affected by color visual deficiency; an example is shown in fig.5. With the Dalton function, the user can select an image, apply the desired type of anomaly, display the result and finally save the obtained image in the original format or in a rotated and scaled version, with reduced width and height.

4.1 COLOR VISUAL DEFICIENCY SIMULATION IN COLORTOOLS

Human color vision is based on three types of retinal photoreceptors, called L, M, and S cones, respectively. The spectral response of such cones depends on the photopigment they contain. Variations of such pigments may shift the cone frequency response, thus changing color sensitivity. This can give rise to a condition usually known as anomalous trichromacy, and specifically to protanomaly, deuteranomaly or tritanomaly, if the cone affected by the shift is L, M or S, respectively. The color vision of anomalous trichromats differs from that of normal trichromats. Color perception is more or less affected, depending on the
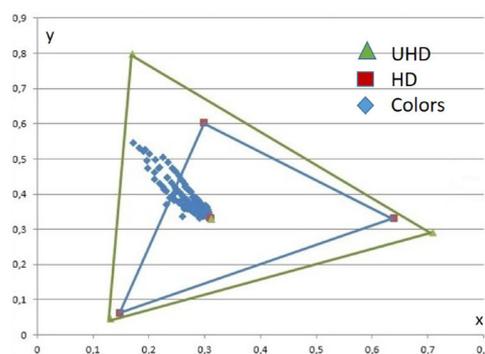
Figure 4 - ColorTools Gamut function.
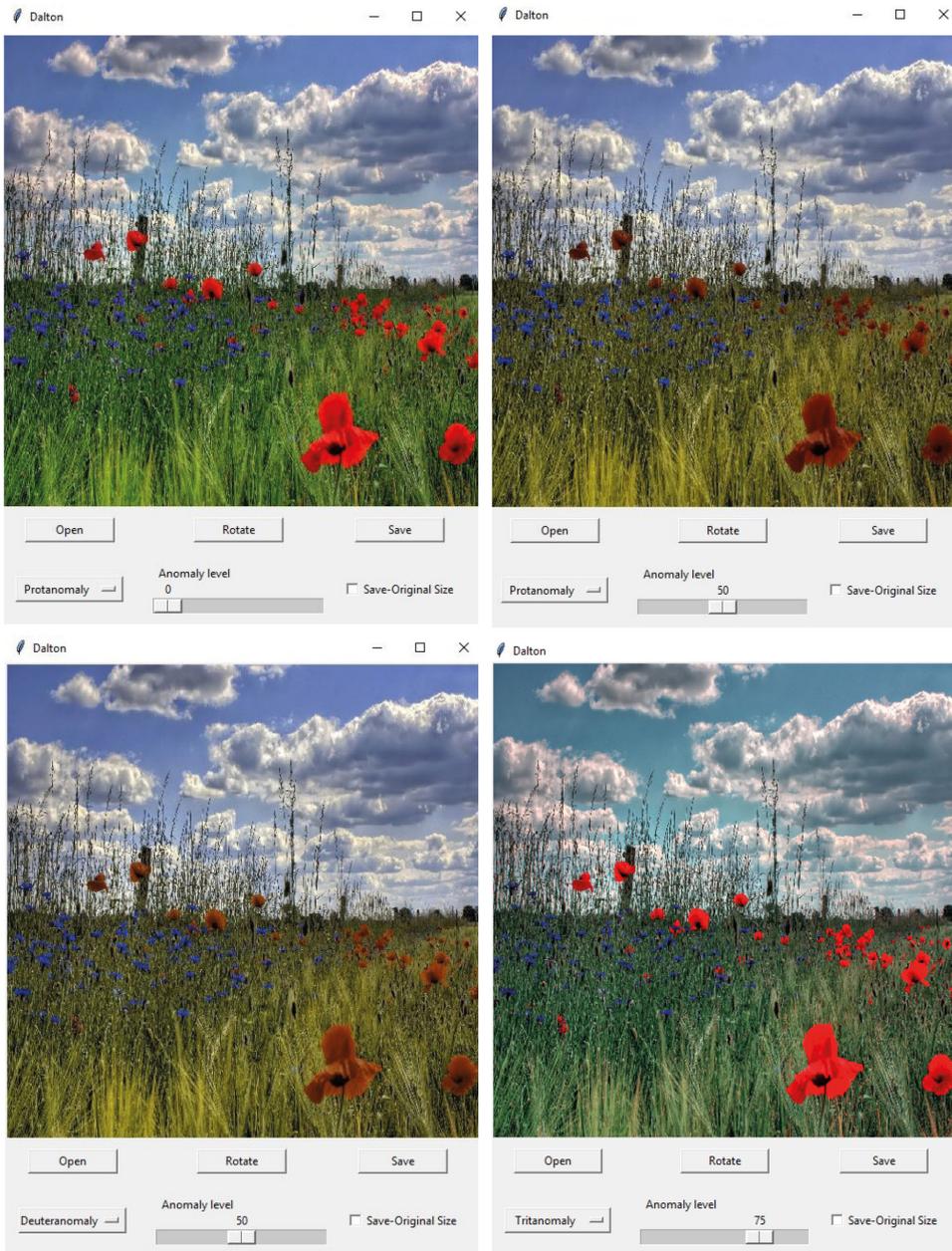
Figure 5 - Dalton function, Color Visual Deficiency is simulated: a) (up-left) normal vision, b)(up - right) protanomaly, 50%, c)(down - left) deuteranomaly (50%), d) (down - right) tritanomaly (50%).

amount of spectral shift suffered by the cone response. Conversely, when one type of cone is entirely missing, dichromacy occurs, i.e. the tridimensional color space is projected onto a two-dimensional plane. Dichromacy is classified as protanopy (L cones missing), deuteranopy (M cones missing) and tritanopy (S cones missing) (Machado, Oliveira, and Leandro, 2009).

In the literature, many algorithms have been proposed to simulate the vision of dichromats. ColorTools has implemented the technique described in (Machado, Oliveira, and Leandro, 2009), which – to the best of the authors' knowledge – is the only one that can simulate not only dichromacy but also anomalous trichromacy. As shown in figure 5, a menu allows the user to select the type of anomalous trichromacy. A slider can change the degree of the anomaly affecting the visual system, in a range from 0 to 100, with 0 corresponding to normal vision, and 100 to dichromacy.

An example of the use of the Dalton functionality is the project proposal for the new Dino Compagni school, in Florence, March 2017. The effectiveness of the color project for visual problems was verified using the Dalton function. As an example, figure 6 shows a schematic project color wheel, used in the proposal, collecting the colors of different internal spaces of the school, showing the colors as seen by normal and anomalous dichromats

5 CONCLUSIONS

The paper has presented *"ColorTools"*, a digital application to support the activities of color designers. The application is written in the Python language. It provides a set of functions, selected from many requirements coming from color designers, to solve some specific problems related to color-science. In the paper, the list of provided functionalities is given and

Figure 6 - Dino Compagni School – Color project of internal spaces, as seen by normal and anomalous thricomats

NORMAL VISION

PROTANOPY

DEUTERANOPY

TRITANOPY
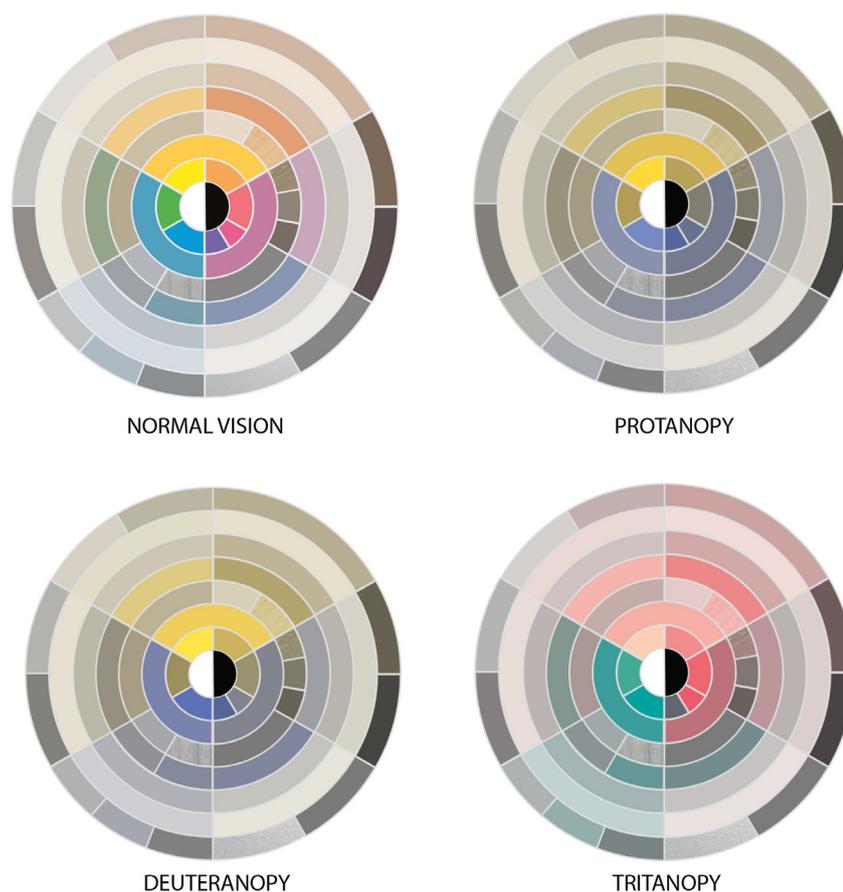
briefly discussed. Also, two particular use cases, related to the problem of color mixing and color vision deficiency in graphical projects, have been presented in more detail. The graphical interface presented by ColorTools to the user has also been presented.

Currently, ColorTools is in a prototypal version, but it has already been successfully tested in some practical applications. Such experiments have shown its utility in color design activities, and for educational purposes in the context of color science.

FUNDING

DECLARATION OF INTEREST

The authors certify that they have no affiliations with or involvement in any organization or entity with any financial or non-financial interest in the subject matter or materials discussed in this manuscript.

BIBLIOGRAPHY

ASTM Standard D 1535-14 (2018), Standard Practice for Specifying Color by the Munsell System, 2008.

Burchett, K.E. (2002), Color harmony, Color Research & Application. 27. 28–31 https://doi.org/10.1002/col.10004.

Cochrane, S. (2014), The Munsell Color System: A scientific compromise from the world of art, Studies in History and Philosophy of Science Part A, Volume 47, 2014, Pages 26-41, ISSN 0039-3681, https://doi.org/10.1016/j.shpsa.2014.03.004.

Color Measurement - Introduction, Historical Perspective, Definitions and Terminology, Components of a Spectrophotometer, Light Source, Detector, Dispersing Element (2018). Available at: http://encyclopedia.jrank.org/articles/pages/1246/Color-Measurement.html (Accessed: 5 October 2018).

Colour Science for Python (2018), available at: http://colour.readthedocs.io/en/latest/index.html. (Accessed: 19 August 2018)

Design Industry; X-Rite (2018). Available at: https://www.xrite.com/industry-solutions/design (Accessed: 5 October 2018).

Landa, E. R., & Fairchild, M. D. (2005). Charting color from the eye of the beholder. American Scientist, 30(5). Available at: http://www.cis.rit.edu/fairchild/PDFs/PAP21.pdf. (Accessed: 3 October 2018)

Machado, G. M., Oliveira, M. M., and Leandro, L. A. F., (2009), A Physiologically-based Model for Simulation of Color Vision Deficiency, IEEE Transactions on Visualization and Computer Graphics. Volume 15 (2009), Number 6, pp.

1291-1298

Munsell Renotation Data (2015) https://www.rit.edu/cos/colorscience/rc_munsell_renotation.php. (Accessed: 19 August 2018).

NCS – We're all about colour (2018). Available at: https://ncscolour.com/ (Accessed: 3 October 2018).

O'Connor, Z. (2010), Colour Harmony Revisited. Color Research & Application. 35. 267-273. https://doi.org/10.1002/col.20578, 2010

Python-Colormath (2014), available at: http://python-colormath.readthedocs.io/en/latest. (Accessed: 19 August, 2018).

Rec. ITU-R BT 2020-2 (2015), Parameter values for ultra-high definition television systems for production and international programme exchange, 10/2015

Rec. ITU-R BT 709-6 (2015), Parameter values for the HDTV standards for production and international programme exchange, 06/2015

Richter, K. (2012), Colour, vision and elementary colours in colour information technology, available at: http://farbe.li.tu-berlin.de/color/ES15.PDF. (Accessed: August, 19th, 2018).

The Munsell and Kubelka-Munk Toolbox (2010), available at http://www.munsellcolourscienceforpainters.com/MunsellAndKubelkaMunkToolbox/MunsellAndKubelkaMunkToolbox.html. (Accessed: 5 October 2018)