



**HAL**  
open science

# Printed image multiplexing by laser processing and its application to security and identity documents

Nicolas Dalloz

► **To cite this version:**

Nicolas Dalloz. Printed image multiplexing by laser processing and its application to security and identity documents. Signal and Image Processing. Université de Lyon, 2022. English. NNT : 2022LY-SES001 . tel-04675353

**HAL Id: tel-04675353**

**<https://theses.hal.science/tel-04675353v1>**

Submitted on 22 Aug 2024

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



N° d'ordre NNT : 2022LYSES001

**THESE de DOCTORAT DE L'UNIVERSITE DE LYON**  
opérée au sein de  
**l'Université Jean Monnet**

**Ecole Doctorale N° 488**  
**Sciences, Ingénierie, Santé**

**Spécialité / discipline de doctorat :**  
IMAGE VISION

Soutenue publiquement le 12/01/2022, par :  
**Nicolas Louis Gabriel Dalloz**

---

**Multiplexage d'images par marquage laser  
pour applications dans le domaine des  
documents de sécurité et d'identité**

---

Devant le jury composé de :

**Ostromoukhov, Victor**  
**Simonot, Lionel**  
**Genevet, Patrice**  
**Garrelie, Florence**  
**Fournel, Thierry**

Professeur des Universités  
Maître de Conférences  
Chargé de recherche EPRT  
Professeur des Universités  
Maître de Conférences

Université Lyon 1  
Université de Poitiers  
CNRS Clermont-Ferrand  
Université Jean Monnet  
Université Jean Monnet

Président  
Rapporteur  
Rapporteur  
Examinatrice  
Examineur

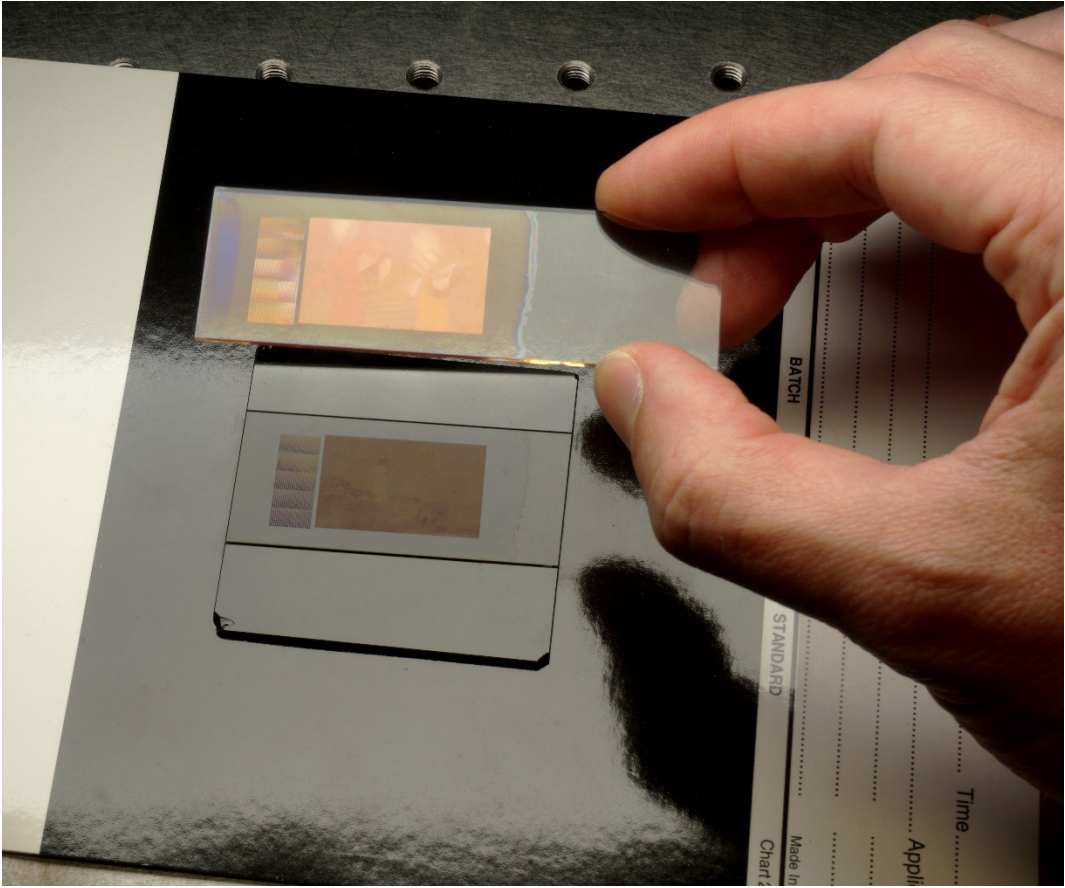
**Destouches, Nathalie**  
**Hébert, Mathieu**  
**Ayala, Stéphane**

Professeur des Universités  
Maître de Conférences  
Responsable scientifique

Université Jean Monnet  
Université Jean Monnet  
HID Global

Directrice de thèse  
Co-directeur de thèse  
Invité







# Abstract

Securing the authenticity of identity documents is a major issue, for which governments and secure documents manufacturers keep inventing more and more advanced security features. However, existing identity documents are often counterfeited, sometimes even a few months after their release. There is a never ending need for new technologies producing striking visual effects, securing documents, and allowing a fast and seamless authentication without the need for any additional device.

Printed image multiplexing, which consists in encoding two or more independent images in the same spot on a material to be revealed under different illumination and observation conditions, is a promising technique to secure documents. Indeed, the multiplexed images are completely entangled, in such a way that it is impossible to falsify one without modifying the others in a noticeable way. However, to this day, few technologies have been shown to enable printed image multiplexing with the characteristics – size, modes of observation – required for an application to identity documents.

By irradiating silver nanoparticles embedded in a mesoporous titania matrix with a femtosecond laser, the Nanoparticles team at Laboratoire Hubert Curien demonstrated the multiplexing of three images. The first image is observed in specular reflection, with the naked eye, while the two others require the use of polarizers. This first result proved that the colors and visual effects produced by these plasmonic metasurfaces can be used for printed image multiplexing. However, as there is not any model able to predict the produced colors from the laser processing parameters, the search for the sets of parameters allowing multiplexing is fastidious. Moreover, the processing times needed to obtain these metasurfaces with a femtosecond are very long, which makes this process unfit for an industrial application.

To overcome these problems and open the way to an application to identity documents, this thesis work focused on creating and implementing a complete and automated process, from the production of metasurfaces, to the printing of multiplexed images, including the search for the color combinations which allow multiplexing. To quicken the processing times, a new nanosecond laser setup with a scanning head is built. To overcome the absence of a predictive model, thousands of metasurfaces are produced by varying the laser processing parameters in small increments. The colors of these metasurfaces are measured in the modes of illumination and observation considered for multiplexing. Thereby, a database linking each set of laser parameters to the produced color in each mode is created. Several algorithms are proposed and implemented to search for metasurfaces which create the color combinations required for image multiplexing in this database.

Several demonstrators were produced, demonstrating two- and three-image multiplexing in a wide selection of modes, some of which are visible with the naked eye without any additional device. One of the multiplexed images can be displayed in full-color thanks to the use of halftoning, an improvement over previous technologies which were only limited to reduced color palettes. The nanosecond laser allows processing speed up to fifty times those obtained with the femtosecond laser. Finally, the methods and algorithms put in place in this thesis are directly applicable to other technologies which produce similar visual effects, thus opening the way to other implementations of printed image multiplexing.

# Résumé

Garantir l'authenticité d'un document d'identité est un enjeu majeur, pour lequel les pouvoirs publics et les imprimeurs de sécurité rivalisent d'ingéniosité en proposant des éléments de sécurité toujours plus sophistiqués. Cependant, les documents existants sont très souvent contrefaits, parfois même quelques mois après leur entrée en vigueur. Il est sans arrêt nécessaire de développer de nouvelles technologies qui permettent d'obtenir des effets visuels marquants, sécurisant les documents et permettant une authentification simple et rapide sans l'aide d'appareils annexes.

Le multiplexage d'image par impression, qui consiste en l'encodage de plusieurs images sur un même support qui sont ensuite révélées sous différentes conditions d'éclairage et d'observation, est une technique prometteuse pour la sécurisation des documents. En effet, les images multiplexées sur le support sont complètement intriquées, de telle façon qu'il est impossible de falsifier l'une d'entre elles sans altérer toutes les autres de façon visible. Cependant, à l'heure actuelle, peu de technologies sont capables de produire des images multiplexées avec des caractéristiques (tailles, modes d'observation) compatibles avec une utilisation pour les documents d'identité.

En irradiant des nanoparticules d'argent incluses dans une matrice mésoporeuse de dioxyde de titane avec un laser femtoseconde, l'équipe Nanoparticules du Laboratoire Hubert Curien a réalisé le multiplexage de trois images. La première image est observée en réflexion spéculaire, à l'œil nu, tandis que les deux autres nécessitent l'utilisation de polariseurs. Ce premier résultat a démontré que les couleurs et effets visuels produits par ces metasurfaces plasmoniques peuvent être utilisés pour multiplexer des images. Cependant, en l'absence de modèle permettant de prédire les couleurs obtenues à partir des paramètres lasers utilisés, la recherche des bons paramètres permettant le multiplexage est fastidieuse. De plus, le temps requis pour obtenir ces metasurfaces avec un laser femtoseconde est très long, ce qui rend ce procédé incompatible avec une application industrielle.

Pour surmonter ces obstacles et permettre une future application aux documents d'identité, cette thèse se propose de mettre en place un procédé complet et automatisé, de la production des metasurfaces à la réalisation d'images multiplexées, en passant par la recherche des combinaisons de couleurs permettant le multiplexage. Pour accélérer les temps d'inscription, un montage laser utilisant un laser nanoseconde et une tête de scanner est mis en place. Pour surmonter l'absence de modèle prédictif, des milliers de



metasurfaces sont produites en faisant varier de manière incrémentale les paramètres de commande du laser. Les couleurs de ces metasurfaces sont mesurées dans les différents modes d'éclairage et observation jugés intéressants pour le multiplexage. Une base de données est ainsi créée, liant chaque ensemble de paramètres laser aux couleurs produites dans chacun des modes. Divers algorithmes sont proposés et implémentés pour rechercher les metasurfaces produisant les combinaisons de couleurs nécessaires au multiplexage d'images dans cette base de données.

Plusieurs démonstrateurs ont été produits, démontrant le multiplexage de deux à trois images, dans une grande variété de modes dont certains observables à l'œil nu sans outil supplémentaire. Il est possible d'afficher une des images multiplexées en vraies couleurs grâce à l'implémentation d'une technique de demi-ton, alors que les précédentes technologies étaient limitées à des palettes réduites de couleurs. Les vitesses d'inscriptions utilisées, grâce au laser nanoseconde, sont jusqu'à cinquante fois supérieures à celle obtenues avec le laser femtoseconde. Enfin, les différentes méthodes et algorithmes mis en place dans cette thèse sont directement applicable à d'autres technologies qui pourraient produire des effets visuels similaires, et ouvrent la porte à d'autres implémentations du multiplexage d'images imprimées.

# Acknowledgements

These last three and half years have been a great journey for me, on both personal and professional viewpoints, through my involvement in both the Laboratoire Hubert Curien and HID Global towards the completion of this thesis. I would like to extend my sincere gratitude to all the people that accompanied me during this period, and who participated directly or indirectly to the work presented in this manuscript.

I would first like to thank my directors, Nathalie Destouches and Mathieu Hébert who supervised this work from the start as a student project during my Master's degree at Institut d'Optique and Université Jean-Monnet, then, as a 6-month internship at Laboratoire Hubert Curien, and of course during the three years of the PhD. Our long discussions, your encouragements, and the time that you managed to carve out of your busy schedule were substantial in steering this work towards what it is today.

I wish to thank Stéphane Ayala, my industrial advisor and manager from HID Global for his continued involvement. You allowed me to dedicate my time to this thesis entirely, while also making me feel like I belong in the company and involving me in supporting several projects of the Credentials Engineering and Innovation team.

I am also grateful to Alice Vermeulin and Marie Wangheluwe who started the collaboration between what was at the time Arjo Systems, now HID Global, and Laboratoire Hubert Curien on the subject of multiplexing, and for recruiting me under a permanent contract with the company, which will allow me to continue working on the subject from the company side after my defense.

Special thanks to Christophe Hubert, who greatly helped with my integration into the laboratory and with the beginning of the experiments with the laser setup, and to my colleagues at the lab, Bálint Elés, Van Doan Le, Manuel Flores and Francis Vocanson. Thanks for a great working atmosphere, interesting discussions, fruitful collaborations, and entertaining lunches. Thanks to Harald Giessen and Florian Sterl for welcoming me in their lab at the 4<sup>th</sup> Physics Institute in Stuttgart, and to Hongfeng Ma, Amaury Habrard and Marc Sebban from Laboratoire Hubert Curien for an interesting exploration of deep learning with the spectral data collected in Stuttgart, which I am sure will be followed by great achievements. Thanks to Lou Gevaux for welcoming me in Newton Technologies in the beginning of this work, and for suggesting looking into dimensionality reduction techniques, which prove very useful in the end of this thesis.

Thank you to all the professors of Institut d'Optique, especially Mathieu Hébert, Raphaël Clerc and Pierre Chavel for their precious teaching and advice.

To Nipun Sharma, thank you for your positive spirit, and for a great team work both when you were still a PhD student and afterwards in HID. Thank you to all the colleagues of the Credentials Engineering and Innovation team as well, for your interest, and your help in discovering many aspects of the ID world.

Many thanks to Aurélien, Raphaëlle, Laure, Greg, Imane, all the Shall we Swing team, and all the Swing dancers in Lyon who welcomed me in this new city and filled my evenings with jazz and happiness.

A warm thank you to Véronique and Jean-Sébastien for hosting me in Paris, and for their continued and joyful support.

Un grand merci à mes parents, Jean-Pierre and Françoise. Merci pour votre amour, votre soutien inconditionnel et vos encouragements. C'est grâce à vous si j'ai pu développer cette curiosité et cet attrait pour l'esprit critique et le doute, si nécessaires à la science, qui m'ont amené jusqu'ici aujourd'hui.

Finally, this thesis could not have been the same without my fiancée Juliette, who fills my days with joy, happiness, and beauty. I could not have accomplished it without your continued support and encouragement, in the good days and the not-so-good ones. You are my guiding light.

# Table of Contents

<b>Abstract</b> .....	<b>5</b>
<b>Résumé</b> .....	<b>7</b>
<b>Acknowledgements</b> .....	<b>9</b>
<b>Table of Contents</b> .....	<b>11</b>
<b>Chapter 1. Introduction</b> .....	<b>15</b>
<b>Chapter 2. Printed image multiplexing</b> .....	<b>21</b>
2.1 <i>Image multiplexing definition</i> .....	21
2.2 <i>Types of image multiplexing</i> .....	23
2.3 <i>Printed image multiplexing in the scientific literature</i> .....	25
2.3.1 Printed image multiplexing via innovative ink layouts	25
2.3.2 Printed image multiplexing via direct plasmonic nanofabrication	29
2.3.3 Printed image multiplexing via laser processing	32
2.4 <i>Conclusion</i> .....	38
<b>Chapter 3. Production of metasurface color databases</b> .....	<b>41</b>
3.1 <i>Laser processing</i> .....	42
3.1.1 Laser set-up	42
3.1.2 Processing parameters	43
3.1.3 Laser processing elements	44
3.1.4 Formed metasurfaces	46
3.1.5 Metasurface database	48
3.2 <i>Multi-mode color measurement</i> .....	49
1.1.1 Optical properties of the metasurfaces	50
3.2.1 Color measurement setup	52
3.2.2 Telecentric lens properties	53
3.2.3 Image processing and color extraction	53
3.2.4 Color gamuts	56

3.2.5 Spectral measurements	57
3.3 Conclusion .....	58
<b>Chapter 4. Production of images .....</b>	<b>60</b>
4.1 Color reproduction workflow.....	61
4.1.1 Image formats	62
4.1.2 Gamut mapping	64
4.1.3 Vector error diffusion halftoning	68
4.1.4 Parameter separation	71
4.2 Experimental results and limitations.....	71
1.1.1 Reproducibility of the colors	73
4.2.1 Interference patterns	73
4.2.2 Pixel overlap	74
4.2.3 Limited pixel count	75
4.3 Conclusion .....	75
<b>Chapter 5. Conditional gamut algorithm.....</b>	<b>78</b>
5.1 Conditions for multiplexing .....	79
5.2 Definitions.....	81
5.2.1 Clustering	81
5.2.2 Hypercolors	83
5.2.3 Conditional gamuts	83
5.3 Conditional gamut algorithm.....	84
5.4 Solutions given by the algorithm and clustering.....	90
5.5 Experimental results.....	92
5.5.1 Two- and three-image multiplexing with naked eye observation	92
5.5.2 Four-image multiplexing using polarized modes	95
5.5.3 Multiplexed elements added to a color image	97
5.6 Conclusion .....	98
<b>Chapter 6. Graph algorithm .....</b>	<b>101</b>
6.1 Graph representation.....	101
6.1.1 Definitions	102
6.1.2 A simple graph representation: 2-mode multiplexing	103
6.1.3 More advanced graph representations: 3-mode multiplexing and more	106
6.2 Algorithms and implementation .....	111
6.2.1 Formulation of the problem	111

6.2.2	MMCE algorithm	112
6.2.3	Implementation	113
6.2.4	Selecting solutions and retrieving the metasurfaces	114
6.3	<i>Experimental results</i> .....	116
6.3.1	2-mode color multiplexing solutions from a synthetic database	117
6.3.2	2-mode color multiplexing solutions from our metasurface database	118
6.4	<i>Conclusion</i> .....	120
<b>Chapter 7. Continuous multiplexing</b> .....		<b>122</b>
7.1	<i>Multimodal halftoning</i> .....	123
7.1.1	Extension of vector error diffusion to multiplexed images	123
7.1.2	Necessity of gamut mapping	124
7.2	<i>Multi-mode gamut mapping: gray scale images</i> .....	125
7.2.1	Multiplexing conditions	127
7.2.2	Optimum	129
7.2.3	Extension to M modes	130
7.3	<i>Multicolor multi-mode gamut mapping</i> .....	130
7.3.1	First case with reduced dimension	130
7.3.2	Change of basis and dimensionality reduction	133
7.3.3	Generalization to $N$ dimensions	137
7.3.4	Application of dimensionality reduction without requiring the use of a device for observation	138
7.3.5	Number of primaries	138
7.4	<i>Experimental result</i> .....	139
7.5	<i>Conclusion</i> .....	140
<b>Chapter 8. Conclusion and outlook</b> .....		<b>142</b>
<b>References</b> .....		<b>145</b>
<b>Appendix A</b>	<b>Laser processing parameters</b> .....	<b>154</b>
<b>Appendix B</b>	<b>Image normalization</b> .....	<b>158</b>
<b>Appendix C</b>	<b>Colorimetry</b> .....	<b>159</b>



# Chapter 1.

## Introduction

Securing the identity of people has been a major concern since the beginning of the 20<sup>th</sup> century. The development and spread of secure passports and identity cards was made possible by the conjunction of, on the one hand, the standardization of offset printing, which enabled to produce official documents with a controlled appearance incomparable with previous handwritten documents, and on the other hand, the development of photography, which tied the documents to their holder in an unambiguous way. Dating back to the 15<sup>th</sup> century, the term “passe-port”, coming from “passe” - going through – and “porte” - the city gates - used to designate a handwritten document, often briefly describing the physical characteristics of the holder, only secured though the use of a specific paper, ink and seal identifying the issuer of the document[1].

For long, the quality and repeatability of the print was a sufficient indicator of the genuine nature of one’s identity document. The tremendous cost and technical knowledge required to operate a security printing press, as well as obtaining the right materials (paper, inks, cover, seal, etc.), was enough to deter counterfeiting attempts. However, with the rapid increase in global exchanges throughout the century, the interest in falsifying passports became greater. Governments and ID documents manufacturers raced to develop and implement new security measures preventing the counterfeiting of identity documents. These security features can be categorized in three levels describing their expected application scenarios :

- Level 1 or “overt” security features are visible with the naked eye and can be checked by any untrained observers.
- Level 2 or “semi-covert” security features, require the use of an additional device such as a UV lamp and/or specific knowledge typically available to trained officers checking passports at an airport counter.
- Level 3 or “covert” security features are checked using expensive laboratory equipment at the manufacturer’s or issuer’s forensic laboratories

With the wide development of digital scanners and printers, making high quality reproductions of documents has become much easier[2]. Part of the answer to this new threat was the implementation of machine-readable documents in the 1980s and RFID



chips since 2000, which helped provide an added digital layer of security. However, these remain subject to security breaches and cannot be solely relied on to prevent identity theft, the cost of forging the “physical” part of the passport remaining the biggest obstacle for fraudsters[3]. Furthermore, these level 2 and level 3 security features can only be checked in a limited number of scenarios, such as during examination at the security checkpoint of an airport, or during an already ongoing criminal investigation, and thus do not prevent the circulation of counterfeit documents as effectively as level 1. Strong level 1 security features are still needed to contain the risk posed by forging and counterfeiting.

Most current level 1 security features prevent copy by making artifacts appear when the document is reproduced, or by adding patterns or images than can be observed only in specific viewing conditions not achieved by common scanners and printers. For example, guilloches and microtext contain thin lines which size are not accurately reproduced with the dotted patterns generated by common digital inkjet printers. Moiré patterns[4], [5], rely on high spatial frequencies to make a watermark appear when scanned at a lower resolution, rendering the document irreproducible. Optical variable elements (OVD) rely on different angles of illumination and observation to display color shifting effects. In this family, special effect pigments[6], iridescent inks and thin-film reflection coatings[7] offer a change of color when the print is rotated. Diffractive optical variable image devices (DOVID) provide even more advanced color shifting properties, by relying on diffractive optical elements created either directly on the document or on a metalized fold. These include two- and three-dimensional holograms, as well as stereograms which induce a perception of movement between two images when the viewing angle is changed.

In parallel, laser marking techniques gained a lot of popularity thanks to their ability to personalize documents with the holder’s portrait and information at the time of issuance. Grayscale laser marking of polycarbonate documents is now the norm. Other derived security features include laser perforation of the holders’ portraits[8], color laser marking by bleaching dyes or pigments[9], and variable laser images obtained through an array of cylindrical lenses embossed into the surface of the card[10]. Their ability to engrave the document through the several layers of its construction makes it very hard to separate from the rest of the document, securing it against counterfeiting. The fast processing times of grayscale laser engraving on centimeter scales and the availability and relatively low cost of the pulsed nanosecond lasers necessary to its application contributed to its popularity. On the contrary, current color engraving techniques are expensive and do not dramatically increase security. Consequently, most documents still contain only a simple grayscale image for the portrait of the holder, which is the most crucial part of the document. Although the trained observer will differentiate it from other printing techniques, for most people, the lack of special visual effect makes it undistinguishable from a high quality print, which fraudsters use as an imitation of laser engraving.

Beside laser engraved images, OVDs have been recognized as some of the best security features to protect a document since the 1990s[2]. They combine the advantage of not being easily reproduced by scanning and printing, and of being overt, with a possibility of adding a “level 2” capability by making them machine readable. However, these features are often applied on separate layers during the document construction. Furthermore, the color shifts it provides are sometimes not easily identified by untrained observers, who do not know what the expected color change is, or which pattern should appear. They thus require wise design choices to protect the holder’s portrait and information[11].

There is therefore an interest in developing new image printing techniques which include several observation and illumination modes, are easily identifiable without the use of any additional device, and able to directly secure the document holder’s portrait.

The Nanoparticle team at Laboratoire Hubert Curien has developed a technology that is a good candidate for this issue. Under laser irradiation, silver nanoparticles embedded in a mesoporous titania matrix coated on a glass substrate are reshaped and reorganized so as to create a wide variety of optical effects when observed under white light. As in certain stained glasses, which also contain metallic nanoparticles[12], the colors of the produced metasurfaces depend on the nanoparticles size distribution and shape. The samples are transparent and reflect light like a mirror. They can be colored with this laser process and have a different color when seen in different observation and illumination conditions, for example against a light source (transmission) or when the light source is reflected on the sample as if it was a mirror (specular reflection). Furthermore, when using polarized laser light for the irradiation, the nanoparticles self-organize into gratings, and additional surface gratings are created when using a femto-second laser[13]. The created gratings diffract light, and they cause the color to be dependent on the polarization under which they are observed, which can be used to create new modes in which new colors can be displayed.

The underlying processes behind the formation of the metasurfaces, as well as the origins of the colors are not yet fully modelled. By carefully selecting the laser parameters, it has been shown that up to three independent images can be encoded on the same spot. One of the images can then be observed in reflection under white light, while the two others are sequentially revealed when the sample is rotated between crossed polarizers. This result of several images encoded on the same spot is called *printed image multiplexing*. As all the images originate from the same metasurface, trying to modify one of the images will directly affect the other ones, thus revealing the attempt of falsification. To produce a multiplexed image, one must know the writing parameters that produce the correct color combinations in the different viewing conditions (which will be referred to as mode hereinafter). The application of printed image multiplexing to secure documents is promising ; however, its industrialization suffers from the low speed of the

technologies with which it was implemented up to now, and from the lack of a proper workflow to implement it in the general case.

The present work proposes to define a framework for the implementation of printed image multiplexing with a nanosecond fiber laser marking set-up. Nanosecond lasers are widely available, including in the ID documents industry, and their longer pulse duration favors thermal effects in the modification of the material. It results in higher processing speeds than the femtosecond or continuous wave lasers, which allows fast laser processing of a centimeter-scale image, compatible with industrial use. We overcome the lack of predictive model for the colors of metasurfaces in the different illumination-observation modes by producing many metasurfaces and characterizing them with a dedicated color imaging system. The set of colors measured on each metasurface in the different modes is called a "hypercolor", and the set of hypercolors measured on the whole set of metasurfaces constitutes a database that we want to explore in order to find the metasurfaces which satisfy the constraints imposed by the multiplexing of independent images, i.e., the possibility to choose the color of each pixel in the image displayed in one mode independently of the colors chosen for that pixel in the images displayed in any other mode. The exploration of the database is one of the main objectives of this work, and three chapters will be dedicated to that problem, in which we introduce various original algorithms. The practical implementation of these algorithms allowed us to produce several demonstrators where up to three images are multiplexed with an industrialization-ready process. Their limits and potential for further extension are also analyzed.

This manuscript is constructed as follows: in Chapter 2, the concept of printed image multiplexing is defined more accurately and a review of the scientific literature on the subject is proposed.

Chapter 3 presents our laser-processing and color measurement workflow, which allows to overcome the lack of a color prediction model with an empirical approach. We create a metasurface database, linking each set of laser parameters to its produced colors in all considered modes, which can then be searched to retrieve interesting colors.

Chapter 4 describes the steps to reproduce an arbitrary color image with these laser-processed metasurfaces. To be printed with our laser setup, the image is first adapted to the range of colors available in the chosen mode. Then, the choice of metasurface to reproduce each of the colors is made via vector error diffusion halftoning. The implementation of this color reproduction workflow for laser-processed metasurfaces allows to adapt and print any color image to be observed in any of the considered modes.

Building on these, three image multiplexing algorithms are presented and implemented, according to three different representations of the printed image multiplexing problem:

Chapter 5 is dedicated to the multiplexing of binary (two-color) images, in two or three modes, with a simple yet effective iterative algorithm. Clustering the nanostructures that display the same color in at least one of the modes reduces the combinatorial complexity of the multiplexing problem and allows finding the right sets of laser parameters among databases containing thousands of nanostructures.

Going further, Chapter 6 leverages concepts from graph-theory to answer the questions “How many images can we multiplex using these metasurfaces?”, “How many colors can have these images?”. Constructing a graph representation of our metasurface databases after clustering allows us to answer these questions by adapting a recently published graph algorithm to color image multiplexing.

Finally, Chapter 7 describes a generalization of halftoning to the multi-mode colors produced by our technology, which allows the multiplexing of full-color images. A geometrical approach, allowing the definition of a continuous “multiplexing gamut” is implemented to optimize the contrast of the images while preventing ghosting effects.



## Chapter 2.

# Printed image multiplexing

The color changing effects of security features are best recognized when the pattern appearing or disappearing is a well-defined image. Everyone will recognize the bird appearing in the VISA hologram, but not every French passport holder could tell what holographic images appear on the data page of their passport when tilted. Although the latter is usually considered more secure than the former because of its advanced color changing effects and high-tech origination process, in practical authentication scenarios, not knowing what color effect should be achieved can prevent the observer from discriminating it from a well-made copy. Instead of mere color shifting, if a security feature shows a transition between well-defined images, that are easily identifiable, making an undiscernible copy will be much more difficult. Further, if the images originate from the same layer of the same material, falsifying one of the images without affecting the others is impossible.

This is the promise of *image multiplexing*, which consists in encoding two or more independent images in the same spot on a material, to be displayed in different illumination and observation modes. From the first appearance of the term in the 1960s, to more recent demonstrations using nanostructures, the potential applications of image multiplexing to security and anticounterfeiting are clear. However, to this day, a thorough implementation of this concept in the secure documents industry is yet to be seen. In this chapter, we will more accurately define image multiplexing and provide the reader with a review of the scientific literature on the subject and in particular on a specific kind of image multiplexing, *printed* image multiplexing, which has the greatest potential for security applications. We will conclude this review by describing what was achieved by our team prior to this work and some of the obstacles that are yet to overcome for its application to secure documents.

### 2.1 Image multiplexing definition

Multiplexing is a signal processing and information theory concept, where several independent pieces of information are combined and conveyed using the same physical channel. In the telecommunication field for example, physical channels can be anything from optical fibers to radio waves. In early types of multiplexing, the multiple signals are

encoded (multiplexed) by modulating one of the parameters of the physical medium, transmitted, and the same parameter is used to decode (demultiplex) each of the transmitted signals. Examples of this include frequency-division multiplexing, which uses various frequencies of carrier waves to encode the various signals, and time-division multiplexing, in which the ability to switch between the different signals over time is used to convey the signals piece by piece. In more recent telecommunication technologies, several parameters can be varied at the same time with the aim of increasing the quantity of information that can be transmitted, sometimes using multiple channels. One example is optical fiber telecommunications where several physical quantities such as polarization, wavelength, or angular momentum can be used simultaneously to multiplex and demultiplex each signal[14].

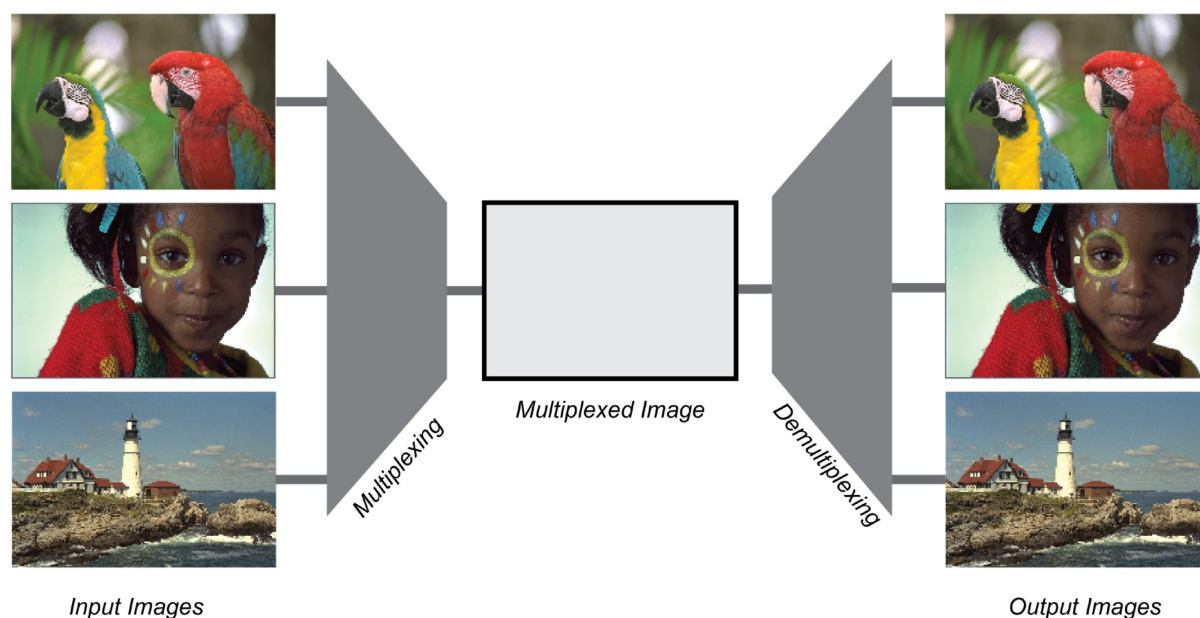


Figure 2.1. Image multiplexing principle illustrated with three images from the Kodak PhotoCD Sampler [15].

*Image* multiplexing is the encoding of two or more independent images onto the same medium. The images are combined into a single multiplexed image, which is written onto the physical medium using one of the processes that will be described in the following section. They are revealed independently in predefined illumination and observation modes: in each mode, the multiplexed image interacts with the incoming light in a controlled way and displays only one of the input images. The particularity of image multiplexing is that the images retain their spatial information throughout the process. The images can thus be read easily with a simple imaging set-up. Before multiplexing, the signals are two- or three-dimensional images, which can each be defined as variations of gray levels (grayscale images) or color (color images) ; in the

multiplexed image, the combined signal is a two-dimensional image where each pixel contains information from all the input images. Finally, the demultiplexing happens by observing the multiplexed image in the predefined modes, and the two- or three-dimensional images are retrieved (Figure 2.1).

## 2.2 Types of image multiplexing

To our knowledge, it was in the carrier-frequency photography field[16], that the term “image multiplexing” was first coined[17]. The principle of carrier-frequency photography is the following: the image of each recorded scene is formed on the photographic plate through an optical set-up performing a spatial filtering. For each exposition, the spatial filter is changed to record the different images with separate spatial frequencies. Then, each image is read by forming the image of the multiplexed photograph through the same optical set-up using the corresponding spatial filter. Spatial filtering is also at the core of many multiple-image printing techniques such as moirés prints[4], lenticular prints[18], or multiple-image prints with reflectors[19], in which a revealing device or layer is superimposed on the print to display the multiplexed images successively. These have been extensively applied for both entertainment and security printing, the most prominent example in secure documents being the CLI/MLI security feature[10]. Although they allow the display of multiple images, and in the case of CLI/MLI of a personalized portrait of the document holder, the images are printed somewhat separately on the medium and not multiplexed at the pixel level, thus allowing their falsification. These techniques are better described by the term “spatial multiplexing” than image multiplexing because the parts of the conveyed images are recorded next to each other. When the spatial multiplexing happens with a period greater than the diffraction limit, which is the case for all cited techniques, this type of prints can be easily characterized and reproduced using commonly available optical microscopes and printing techniques.

Another type of image multiplexing is *holographic* image multiplexing, or hologram multiplexing, where different holograms are projected out of the material plane through phase and amplitude manipulation. Leith and Upatnieks[20] proposed the first holographic image multiplexing scheme where two transparencies are placed side by side and recorded in the same hologram. The two transparencies are placed at different distances from the holographic plate without any overlap. Their hologram is recorded using the classical interferometric exposure method. The demultiplexing can be performed by imaging the corresponding parts of the hologram. In a later paper, Leith et al.[21] describes another method where the angle of the reference beam is varied to record several holograms, one at each reference beam angle, in thicker holographic plates. This method is still used to produce color holograms and animated holograms and has been explored for data storage, although with little commercial success[22]. A third



major way of performing hologram multiplexing is also related to spatial filtering. Caulfield[23] and later Som and Lessard[24] proposed spatial filtering schemes to be applied during the exposure of the holographic plate, enabling the recording of several images on different parts of the plate, which allow a subsequent demultiplexing by applying the same spatial filtering. As with regular single-image holograms, the literature on holographic image multiplexing evolved towards computer-generated holograms that can now be created through nanofabrication techniques such as electron beam lithography (EBL)[25]. Recent advances in this field demonstrate high storage capacity with an encoding which is based on wavelength[26]–[30], polarization[30]–[35], incidence angle[36], spatial frequency[37] or angular momentum control[38] to switch between the multiplexed images. It also allows full color holographic patterns[39], [40]. The design of these multiplexed holograms is done with well-known computer-generated holograms algorithms and is well described in [41].

The last type of image multiplexing is *printed* image multiplexing, where the multiplexed images are displayed on the material itself through the control of the color of each pixel in the different modes. In this form of multiplexing, each pixel of the multiplexed image directly encodes the colors of the corresponding pixels in the demultiplexed images. Printed image multiplexing relies on the ability of the printing material to display a different color in each of the modes. Such color changes can be created by using specific inks such as metallic[42], or fluorescent inks[43], [44], printing on metallic substrates[45], or recto-verso prints observed in transmission[46]. It can also be achieved by the precise control of the electromagnetic field at a nanoscopic scale to produce the desired color change. Such control can be obtained with plasmonic metasurfaces, which are metal-dielectric nanostructured surfaces producing uncommon optical effects thanks to local field enhancement. Under monochromatic light, a high number of images can be multiplexed with plasmonic metasurfaces, using parameters such as wavelength, incidence angle and polarization for the demultiplexing[47]–[51]. Under white light however the number of multiplexed images is limited to two or three[52]–[55] and always requires the use of polarizers for demultiplexing. Most demonstrations of image multiplexing with plasmonic metasurfaces involve rather low-throughput and complex nanofabrication processes such as EBL and focused ion beam (FIB) for the fabrication of the metasurfaces, except a recent work from our group which used femtosecond laser to process the metasurfaces[54]. This kind of image multiplexing holds great promise for security applications and will be the focus of this work.

Printed image multiplexing is the best candidate for security printing. As discussed above, spatial multiplexing schemes are subject to reproduction by any printing technique as soon as the spatial filter can be removed from the document. On the other hand, more and more counterfeiters are able to produce holograms. Provided that one masters a way of producing holograms and can reconstruct a specific hologram correctly, one can produce an identical copy of this hologram: contrarily to printed images where

the printed colors and general aspect of the print depend on the printing technique, a hologram records all the information of the light diffracted by the original object. When it is reconstructed, it thus provides all the information necessary to produce an identical copy with a similar holographic process. Differentiating the hologram processes requires thorough characterization with expensive laboratory equipment, for example with scanning electron microscope (SEM). On the contrary, colors from a printed image have a specific visual aspect inherent to the printing processes, which can often be authenticated by visual observation. Furthermore, although color holograms and holograms observable under white light do exist, hologram multiplexing remains confined to monochromatic images with coherent observation, which is not possible with level 1 observation. Printed image multiplexing is thus the only kind of image multiplexing that can be used as a level 1 security feature while providing an intrinsic protection against tampering.

### **2.3 Printed image multiplexing in the scientific literature**

Several demonstrations of printed image multiplexing exist in the literature. Various technologies, from ink-based printing with special-effect inks or metallic substrates to the manipulation of nanostructures with lasers, have been used in that aim and are described in the following sections.

#### **2.3.1 Printed image multiplexing via innovative ink layouts**

Printing inks on paper is one of the oldest ways of reproducing images. In standard applications of printing, the printing support is strongly scattering, and the inks are absorbing and virtually non-scattering. This produces colors which are almost entirely independent from the illumination and observation conditions such as the incidence angle of light on the print. However, with the use of special effect inks, special printing substrates such as metallic films, or recto-verso printing, one can produce colors than are dependent on the angle of illumination and observation (goniochromatic) or which change when the print is held against a light source. Most of these effects are already known in the security printing industry and used as color-changing security features. This is the case for optical variable inks (OVI) which color changes with the direction of illumination and observation, and the feature called “see-through register”, in which prints on both the recto and verso sides of a passport page align when held against a light source. However, few examples in the literature demonstrate actual image multiplexing, where uncorrelated images are formed in each of the modes. We describe and comment them in the following.

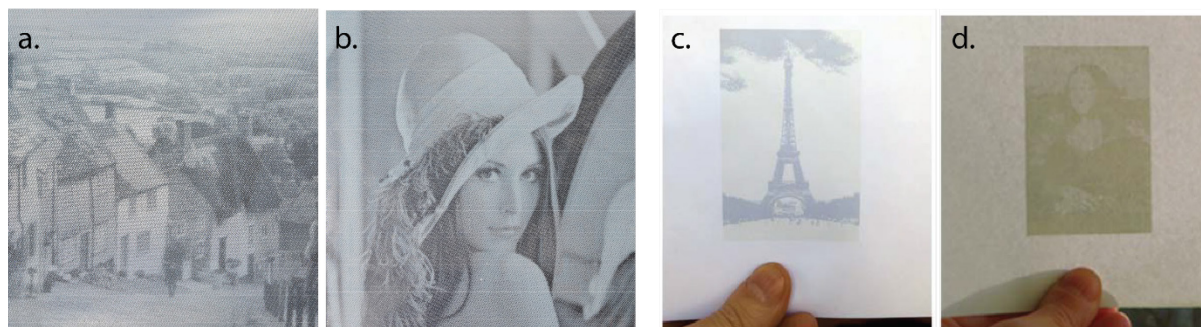
Pjanic and Hersch[45] controls the goniochromatic properties of prints obtained with classical inks on a silver substrate by adding a fourth white ink to the cyan magenta yellow (CMY) inks. They consider two modes: “specular reflection” and “diffuse

reflection". In mode "specular reflection", the print is observed in the direction where the light coming from the light source would be reflected if the sample was a mirror. In this mode, if the sample was a mirror, the observer would see the image of the light source. In diffuse reflection, the direction of the incident light is normal to the surface of the print ( $0^\circ$ ) and the print is observed at a  $45^\circ$  angle. In this mode, a mirror would appear black, because all of the light is reflected in the normal direction, and none is reflected in the direction of observation. However, if the ink is scattering, part of the light can be scattered towards this direction of observation: this corresponds to the conditions under which we observe classical prints on paper.

As mentioned above, classical CMY inks are non-scattering. When printed directly on a metallic substrate, they give bright colors when observed in specular reflection: most of the incident light is directly reflected by the metallic substrate in the specular direction. In diffuse reflection, only a small part of the light is scattered in the observation direction and the print looks dark. When white scattering ink is printed under the CMY inks however, the light reflecting on the white ink is scattered in many directions. In specular reflection, the amount of light reaching the observer is smaller because part of the incoming light has been scattered away from the specular direction. On the contrary, in diffuse reflection, part of the light is scattered in the observation direction, making the print appear lighter. By controlling the amount of white and CMY ink, one is therefore able to control the lightness of the print in specular and diffuse reflection: a higher quantity of white ink makes the print darker in specular reflection and lighter in diffuse reflection, a higher quantity of CMY inks makes the print darker in both modes due to absorption. The authors calibrate a spectral prediction model to predict the color of these prints made on metallic substrates. By inverting this spectral prediction model, they demonstrate three different results. First, they identify all ink combinations that provide a color that do not change from one mode to the other, to print images that are bright and colorful in both modes. Then, they identify those colors which lightness can be tuned in one of the modes by increasing/decreasing the amount of white and CMY inks and use them to embed patterns that are revealed by a lightness variation in one of the modes. Although the hue and chroma of the image is unchanged, the lightness variations they achieve make a pattern appear, as a watermark. Finally, they perform image multiplexing, by using the lightness variations obtained in both modes to display two uncorrelated grayscale images (Figure 2.2a,b).

In this work, the authors benefit from the large literature on spectral prediction models for prints with CMY inks. By adapting the classical printing workflow to a new substrate material, adding a white scattering ink, they identify two modes, specular reflection, and diffuse reflection in which they are able to obtain interesting optical effects and to perform two-image grayscale multiplexing. The direct link between the quantity of white ink and the lightness in each of the modes, as well as the availability of a spectral

prediction model, makes it possible to easily identify ranges of ink quantity in which multiplexing is made possible by finding a trade-off between the contrast of both images.



*Figure 2.2 Ink-based printed image multiplexing (a,b) image multiplexing produced by printing inks on a metallic substrate, the first grayscale image (a) appears in specular reflection while the second image (b) appears in diffuse reflection, reproduced from [45] (c,d) image multiplexing produced by recto-verso printing, the first image (c) appears in diffuse reflection while the second (d) appears in transmission, reproduced from [46].*

In a subsequent work [56], the same authors print inks in a crossed lines pattern. Most inkjet printers are operated in an all-or-nothing manner: for each ink and each point of the print, the printer either projects a droplet or no droplet, and is not able to directly control the amount of ink. To produce continuous color variations, halftones are usually created by alternating inked and blank areas to form color dots of varying size. With line halftones, inked areas are organized to form lines of varying thicknesses instead of dots. Contrary to dot halftones, lines halftones are anisotropic. This creates anisotropic prints when printed on a metallic substrate: to make the metallic substrate printable, an ink-attracting polymer layer is deposited on top of the metallic surface. When light is entering the print with an angle, the light is reflected by the metallic surface and goes out of the print further away with a distance depending on the thickness of the polymer layer and the angle of incidence of the light. Depending on the orientation of the incoming light, light entering the print through one of the inked areas might come out in an area which is not inked, and inversely. The lines that are perpendicular to the direction of observation therefore appear wider than the lines which are parallel to it, resulting in a larger effective ink coverage. When the sample is observed in specular reflection, rotating the sample by  $90^\circ$  changes the way light interacts with these halftone lines, which in turn changes the color displayed by the print. By crossing two line halftones and controlling the ink coverages of each halftone, the authors are able to obtain two different colors in the two modes formed by this  $90^\circ$  rotation. In the article, they produce several demonstrators where colored parts of the print in one mode are either desaturated or recolored in the other. The authors obtain these effects by optimization of the ink amounts corresponding to each of the CMY inks for each line orientation (six values in

total). Although the color variation obtained can be huge, the optimization can result in the creation of a ghost images, as a modification of the color in one mode also modifies slightly the color in the other mode. The authors cite this as a limitation and have chosen to limit the examples shown to “simple” color shifting effects because of this ghosting effect. However, one demonstrator of two-image multiplexing using this technique have since been exhibited at the 2019 Color Imaging Conference, showing two different color images for the two orientations, although with rather unsaturated colors. No method is given for the production of multiplexed prints showing two arbitrary images with this technology.

Another possibility to obtain image multiplexing with traditional inks is creating recto-verso prints (also known as duplex prints) by printing inks on both sides of the paper. The use of duplex prints to convey two separate pieces of information to be observed in reflection from both side of the paper is of course trivial. However, if the duplex print is observed in transmission, the ink on both sides of the paper interacts with light at the same time. By controlling the ink amounts carefully using a spectral prediction model, Dalloz and Hébert [57] created demonstrators of two-image multiplexing in modes reflection and transmission (Figure 2.2c,d). Here, six parameters are available to the designer of the print : CMY ink amounts on the “recto” side of the print, which create the image observed in reflection alone, and CMY ink amounts on the “verso” side of the print, which interact with light only in the transmission mode. As CMY inks create color by absorbing light, one obvious limitation is that once the reflection image is printed, one can only “add absorption” by printing inks on the verso. If we print black on the recto side of the print, this point will also appear black in transmission, and adding ink on the verso cannot change it. Therefore, a trade-off must be found between the maximum amounts of inks used to reproduce the color variations in the image seen in reflection (printed on the recto side) and the amounts of inks needed on both sides to reproduce the image seen in transmission. A spectral prediction model adapted to predicting the reflectance and transmittance of duplex prints is calibrated and used to compute the ink amounts needed to display the desired image in transmission once the image displayed in transmission has been chosen. Because of the limitation discussed above, the colors used for this multiplexing are rather pale.

Although traditional ink-based printing benefits from well-known and efficient spectral prediction models, the smooth color variations it provides do not yet allow the clear decorrelation of one mode from another. This can be seen in the limitations observed by the authors of these works: in the first one, only the lightness of the prints can be controlled independently, in the others, the modes are not decoupled enough and trying to obtain a big color variation by modifying the inks quantities will result in a ghost image. A couple of other works provide more freedom by relying on different light sources and specific inks, which allows the multiplexing of two images. In one, the control of the amount of black ink in the color separation of security prints allows displaying a

color image under visible light and to form a grayscale image on a detector under infrared illumination[44], in the other, printing with specific red green blue (RGB) UV fluorescent inks allows multiplexing two color images under visible and UV illumination[58]. However, in a security feature application, these light sources are not available in every scenario and can thus prevent the observer from verifying the document. Under white light, to produce a striking color image multiplexing, more control should be available over the way the light interacts with the print. This can be achieved with metasurfaces.

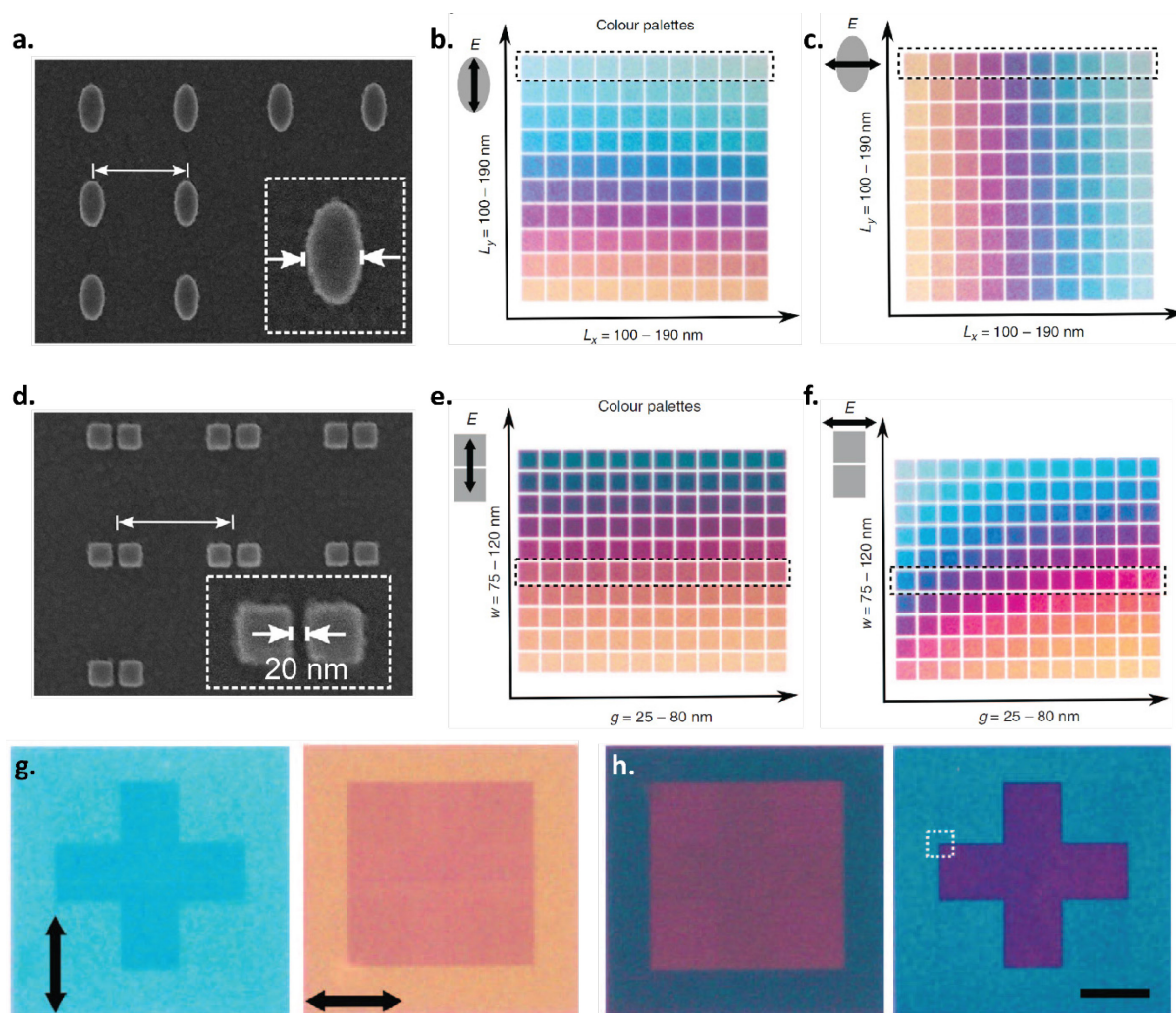
### 2.3.2 Printed image multiplexing via direct plasmonic nanofabrication

Photonic metasurfaces are man-made nanostructured arrays of materials, which can include both metals and dielectrics, and are specifically designed to modify the amplitude, phase, and polarization of the incoming light. Among these, plasmonic metasurfaces are constructed by metallic meta-atoms. They provide strong field enhancement through the excitation of surface plasmon resonances under visible light: when light is shone unto these metallic nanostructures, collective oscillations of conduction electrons at the surface of the metal are excited and give rise to unique optical properties. In particular, plasmonic metasurfaces have been studied as a way of engineering visible light into structural colors[59]. They have been found to produce highly saturated and possibly environmentally friendly colors. Further, they can create polarization dependent colors, which have been used for color image multiplexing. “Top-down” approaches such as electron beam lithography (EBL), focused ion beam lithography or milling provides unrivaled precision in manipulating these nanostructures, which in turn allows the precise control of their electromagnetic properties.

Goh et al. [52] use the polarization sensitivity of metallic nanoposts of ellipsoidal (Figure 2.3a) or square section (Figure 2.3d), to produce polarization-multiplexed images. EBL allow the authors to precisely control the geometrical parameters of the nanoposts: for ellipsoidal nanoposts, the axes length  $L_x$  and  $L_y$  are varied independently ; for square nanoposts two squares of width  $w$  are put together and spaced by a gap  $g$ . The ellipses and pairs of nanosquares are regularly distributed on the substrate with a period of 100nm. The produced nanostructures are observed with a bright-field microscope in reflection, to which a linear polarizer is added. Two modes are defined by aligning the axis of the linear polarizer with either one of the axes of symmetry of the nanoposts. The scattering properties of the nanostructures are affected by their dimensions along the selected axis of polarization, thus changing the color they display. In the case of ellipses, a variation of their axis length corresponding to the incident polarization directly changes their color in the corresponding mode and not their color in the other mode where the polarization is orthogonal (Figure 2.3b,c). This is ideal for multiplexing: the two parameters have an independent role on the production of the color in each mode, thus the two images can be encoded independently by varying only one of the parameters for

each image. For nanosquares however, the relation between the geometrical parameters of the nanostructures and their color in both modes is less straightforward. When the polarization axis is aligned with the two nanosquares, only the width of the nanosquares has an impact on the color (Figure 2.3e). However, their color depends on both parameters in the orthogonal mode (Figure 2.3f). Nonetheless, the authors demonstrate two image multiplexing with both structures. Although, as discussed above, the design of the images for the ellipses case is straightforward, the authors do not report how they selected the parameter values to encode the images made with nanosquares.

Using a combination of metal-evaporation, electron-beam lithography, reactive ion etching, and inductively coupled plasma deposition, Heydari et al. [53] create cross-shaped holes in an aluminum layer. Similarly to the previous work, their structures provide two obvious geometrical parameters: the respective arm-length of the crosses. This enables them to tune the displayed color of their nanostructures in two polarized transmission modes where the axis of the polarizer is set parallel to either arm of the crosses. However, to maintain a high transmission while creating strong color changes, the authors chose to vary both the axis length and the spatial periods at which the crosses are repeated in both directions. This complexifies the identification of independent parameters controlling the color in only one of the modes. Although the design process is not detailed thoroughly in the article, the authors demonstrate the multiplexing of two images containing a large palette of saturated colors. However, the demonstrator is made of homogeneous patches of color and is not representative of an arbitrary color image, such as one obtained from a digital camera.

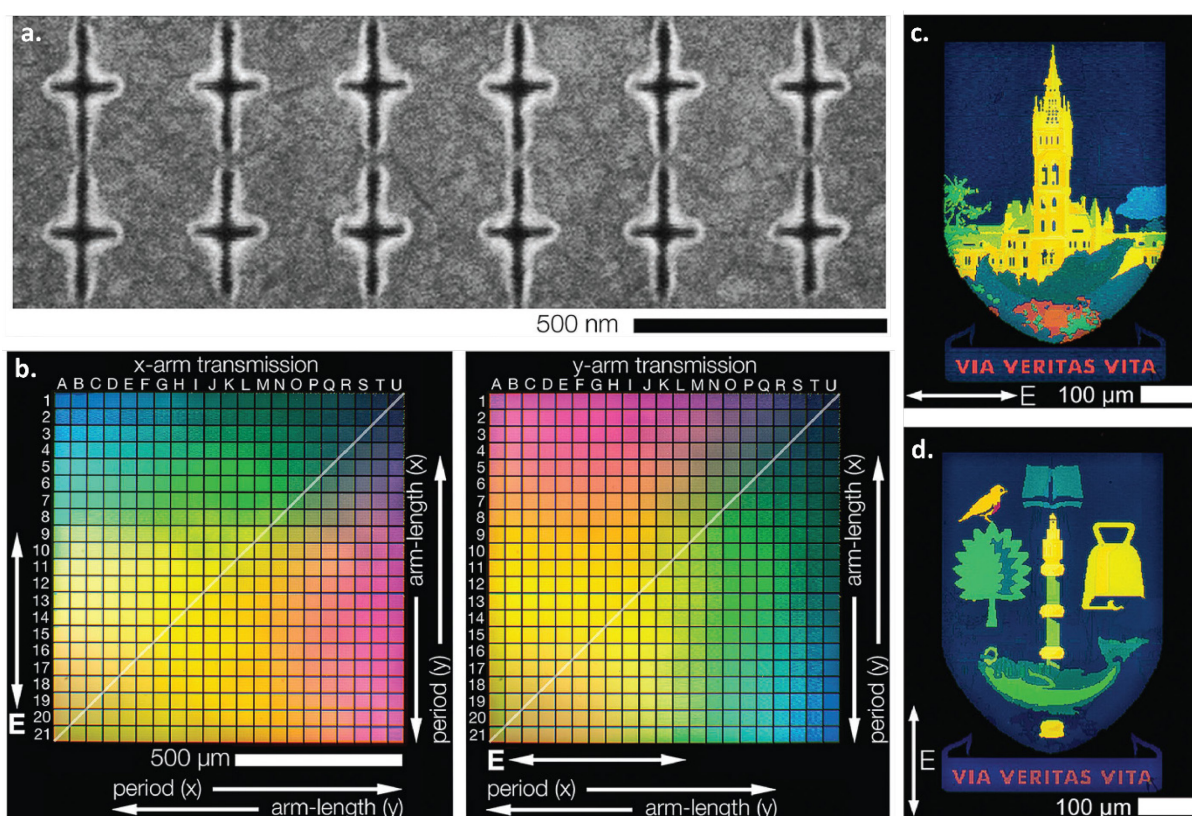


*Figure 2.3 Two types of nanoposts allowing two-image multiplexing with orthogonally polarized reflection modes (a) SEM image of the ellipsoidal nanoposts (b) colors reflected under light polarized parallel to the longest axis of the ellipses (c) colors reflected under light polarized perpendicular to the longest axis of the ellipses. (d) SEM image of the square nanopost doublets (e) colors reflected under light polarized parallel to the axis passing through both nanosquares (f) colors reflected under light polarized perpendicular to that axis (g) multiplexing result with ellipsoidal nanoposts, the double arrow shows the axis of polarization of the light (h) multiplexing result with nanosquares in the same modes. Length of the scale bar is  $20 \mu\text{m}$  and is the same for all multiplexed images (g,h). Reproduced from [52]*

These articles demonstrate that precisely constructed arrays of metallic nanostructures allow image multiplexing by varying well defined geometric parameters. Two-image multiplexing can be achieved under white light with at least one polarizer, by identifying polarizations which are only affected by one of the parameters. However, the low number of available design parameters for these geometric structures, and the lack of a general method makes the extension of this multiplexing to other modes, or more



images, difficult. Further, the fabrication processes needed to achieve the nanoscopic precision required to produce the desired resonances require many steps and have a low-throughput. Although they possibly can be reproduced “en masse” with nanoimprint lithography, this does not allow for personalization and is limited to the reproduction of a predefined master. On the opposite, some laser processes have also arisen for the production of plasmonic structures, which are flexible and allow the processing of larger areas faster. These have been used to produce color images, and more recently the multiplexing of three images by our team. The next section focuses on such processes.



*Figure 2.4 Multiplexing realized with cross-shaped plasmonic structures. (a) SEM image of the cross-shaped nanostructures (b) colors obtained by varying the geometrical parameters under light polarized along each axis (c, d) demonstrator showing multiplexed images in transmission with light polarized along each axis. Reproduced from [53]*

### 2.3.3 Printed image multiplexing via laser processing

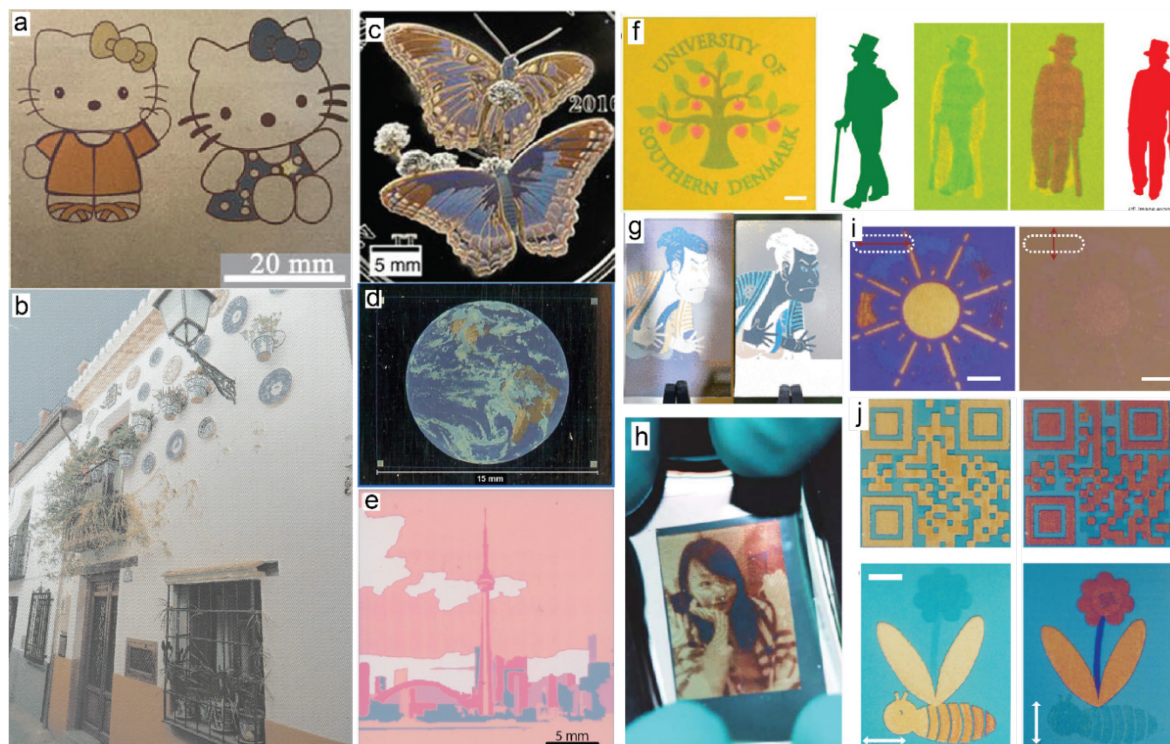
Laser processes are a promising technology for the personalized printing of color images. Compared to previously described nanofabrication techniques, they provide high throughput comparable to inkjet printing, while creating a full range of colors and optical effects. Further, laser processes are able to produce colors below the surface layer of the printed substrate, which has interest for both durability and security. Grayscale image marking gained a wide popularity in identity documents due to its high throughput,

contrast, and availability of the lasers used to process polycarbonate. As mentioned in introduction of this work, several strategies have already been developed for laser color image printing on identity documents, based on the selective bleaching of pigments previously deposited in layers or printed on polycarbonate sheets[9]. However, the process requirements of these two technologies (several laser wavelengths for one, camera based-registration between the laser and the inks for the other) result in expensive processing set-ups and relatively low throughput compared to single laser schemes such as grayscale personalization. Further, they are both based on absorbing pigments deposited over a white diffusing substrate, which produce diffusing colors akin to inks on paper and thus are not interesting for our printed image multiplexing. Nonetheless, several other laser processing strategies are reported in the literature, with a high potential for both colorful images and special optical effects, including printed image multiplexing. We will review these shortly and describe a recent achievement of our team towards color image multiplexing.

Four main processes have been described for color laser marking. The first one is the most well-know and relies on the creation of oxide films on top of metals such as stainless steel or titanium[60], [61]. The colors obtained with such processes are highly resistant to both environmental conditions and mechanical impacts [62]. Upon nano, pico or femtosecond laser irradiation, oxide layers form on top of the metal, creating a layered structure which generate colors by thin film interferences[63]. The colors are angle dependent [64] which could be of interest to generate printed image multiplexing, however we are not aware of any realization in this sense. An expensive literature exists on this process, often demonstrating colored cartoon drawings or logos to illustrate the marking of color images (Figure 2.5a). However, using this process, one recent work by Cucerca et al.[65] present what we think is the most advanced color reproduction workflow described for the laser processing of a color image until now. They reproduce any arbitrary color image by selecting a few sets of laser parameters producing color primaries which are used in line halftones (Figure 2.5b). Unfortunately, the angle sensitivity of the prints is observed, but not used as a potential security feature. No printed image multiplexing has been demonstrated with this process.

The second kind of color laser marking relies on the creation of nanoparticles on bulk precious metals[66], [67]. A pulsed laser is shone onto a piece of bulk metal, which creates nanoparticles by ablation. Bright colors are formed even on non-planar surfaces, resulting in visually appealing coloration of jewelry or silver coins (Figure 2.5d). Due to the random distribution of nanoparticles these colors are not angle dependent. This property is highly desired for decorative applications but prevents its use for multiplexing since angle dependency would be the only possibility to allow different observation modes on bulk metals. On the contrary, a third laser process used for color laser marking is the generation of light induced periodic surface structure (LIPSS). These

create surface gratings which produce colors due to diffraction, resulting in highly iridescent colors. Dusser et al. [68] characterize and use these structures in a color image in the fixed observation conditions of a scanner (Figure 2.5d).



*Figure 2.5 Single and multiplexed color images by laser processing (a) cartoon image printed on stainless steel by laser-induced oxidization, reproduced from [63] (b) arbitrary full-color photograph printed on stainless steel with the same process and full color reproduction workflow, reproduced from [65] (c) color drawing on silver coins due to nanoparticles formed by ablation, reproduced from [66] (d) color image drawn with LIPSS, reproduced from [68] (e) color image produced by continuous CO<sub>2</sub> laser scanning of gold thin films, reproduced from [69], (f) color drawing, and multiplexed print showing two different silhouettes in two different polarization states, reproduced from [70] (g) color artwork reproduced by laser controlled dewetting of Ag thin films observed in transmission (left) and specular reflection (right), reproduced from [71], (h) arbitrary color image obtained by resonant laser printing on Al nanoposts, reproduced from [72] (i) color drawing and (j) two polarization dependent multiplexing demonstrators obtained by resonant laser printing on Al nanocrosses, observed with light polarized vertically (left) or horizontally (right), reproduced from [73]*

Finally, laser irradiation can also be used for the local tuning of the optical properties of already created nanostructures. The nanostructures can be created with the full range of nanofabrication techniques, from “bottom-up” methods such as thin film deposition to “top-down” methods such as nanoimprint lithography. Although the precise control of individual nanostructures with nanofabrication techniques such as EBL is slow and expensive, several cost-effective strategies exist for replicating identical nanostructures

on wide areas. By relying on subsequent laser irradiation for the modification of individual nanostructures, color images can be created, sometimes with sub-wavelength resolution, creating colors and optical effects comparable with previously discussed direct nanopatterning techniques, with much higher processing speeds. A first substrate used for such methods are semicontinuous metal films (SMF) consisting of random nano-islands obtained by deposition techniques such as pulsed vapor deposition (PVD). Ooms et al. [69] generate colors by irradiating a thin film of gold deposited on glass with a continuous laser, resulting in the localized heating and annealing of the gold films. They demonstrate the interest of this technique for image printing by inscribing a large color image (6 cm<sup>2</sup>), with a resolution of 180 μm (Figure 2.5e). Roberts et al. [70] uses a Au/SiO<sub>2</sub>/Au structure in which the top layer is a ultrathin gold layer deposited by electron-beam evaporation. Irradiation by a femtosecond laser induces a melting and reshaping of the nano-islands which they use to create polarization dependent colors. Although not discussed in the article, the authors produce a polarization-multiplexed printed image using this technique, which is shown in the Supplementary information of the article and reproduced here (Figure 2.5f). However, significant cross-talk is visible and creates ghost images. Kuroiwa et al. [71] proposes the controlled laser dewetting of a thin Ag film with an infrared nanosecond laser. This process, while using a relatively low cost and widely available laser, produces interesting transmission-scattering dichroic colors: the created metasurfaces have a different color when observed in transmission and in diffuse reflection (a mode which is called scattering here). This produces an appealing optical effect on the produced color images (Figure 2.5g) but this effect is not used for multiplexing. Laser-induced reshaping can also be performed on more precisely constructed arrays of nanostructures: identical and regularly spaced nanostructures are replicated on large surfaces, for example by nanoimprint lithography. Then, a femtosecond or picosecond laser is focused on the nanoposts. Due to the resonance properties of the nanostructured array, a highly localized temperature rise is generated at the center of the beam, allowing to modify the nanostructures with a sub-diffraction precision. Zhu et al.[72] creates a structure consisting of polymer nanoposts on which an Al thin film is deposited (similar to the previously cited work by Goh et al. [52]) only here the anisotropy of the structures is subsequently created by subjecting the circular nanoposts to femtosecond laser pulses. By operating at laser powers close to the melting threshold of Al, the authors are able to address single nanoposts, resulting in a 127 000 DPI resolution. Although vibrant colors are achieved in homogeneous color squares, the authors' attempt at reproducing a color image by halftoning results in a rather colorless image (Figure 2.5h). They acknowledge the polarization sensitivity of the colors (Figure 2.5i) but do not demonstrate image multiplexing. An natural extension of this method is presented by Zhang et al. [73] this time modifying cross-shaped Al nanostructures formed on top of an Al layer with a silica spacer (structure similar to the previously cited work by Heydari et al. [53]). Color images and printed image

multiplexing based on polarization (Figure 2.5j) as well as holographic multiplexing is achieved with this technique. Although these laser processes are not only focused on printed image multiplexing, they show the possible use of previously described structures on larger scale with much more versatile and cost efficient processes.

The laser processing method implemented by our team can also be regarded as the laser tuning of plasmonic metasurfaces. As described in several previous articles[13], [74]–[76], continuous or femtosecond laser irradiation of silver nanoparticles embedded in a mesoporous titania matrix create plasmonic colors exhibiting both polarization and transmission-scattering dichroism, as well as diffraction from both embedded nanoparticle gratings and surface gratings. The samples are prepared by impregnating a sol-gel deposited mesoporous titania film with silver salts, followed by UV-Visible light cycles designed to grow small silver nanoparticles in the mesoporous matrix. Then, under laser irradiation, the nanoparticles grow and self-organize, which results in metasurfaces which nanoparticle size distribution and spatial organization depends on the power, repetition rate, polarization and scanning speed of the laser (Figure 2.6d). In a recent article [54], we describe the concept of finding logical color trees for identifying the nanostructures which enable printed image multiplexing (Figure 2.6a,b). Our approach is to find a set of nanostructures which provide the color combinations needed to reproduce any binary image, rather than selecting the parameters in an image dependent manner. We describe geometrical patterns that helped graphically explore and verify multiplexing solutions by inscribing and visually checking them for any ghosting issues.

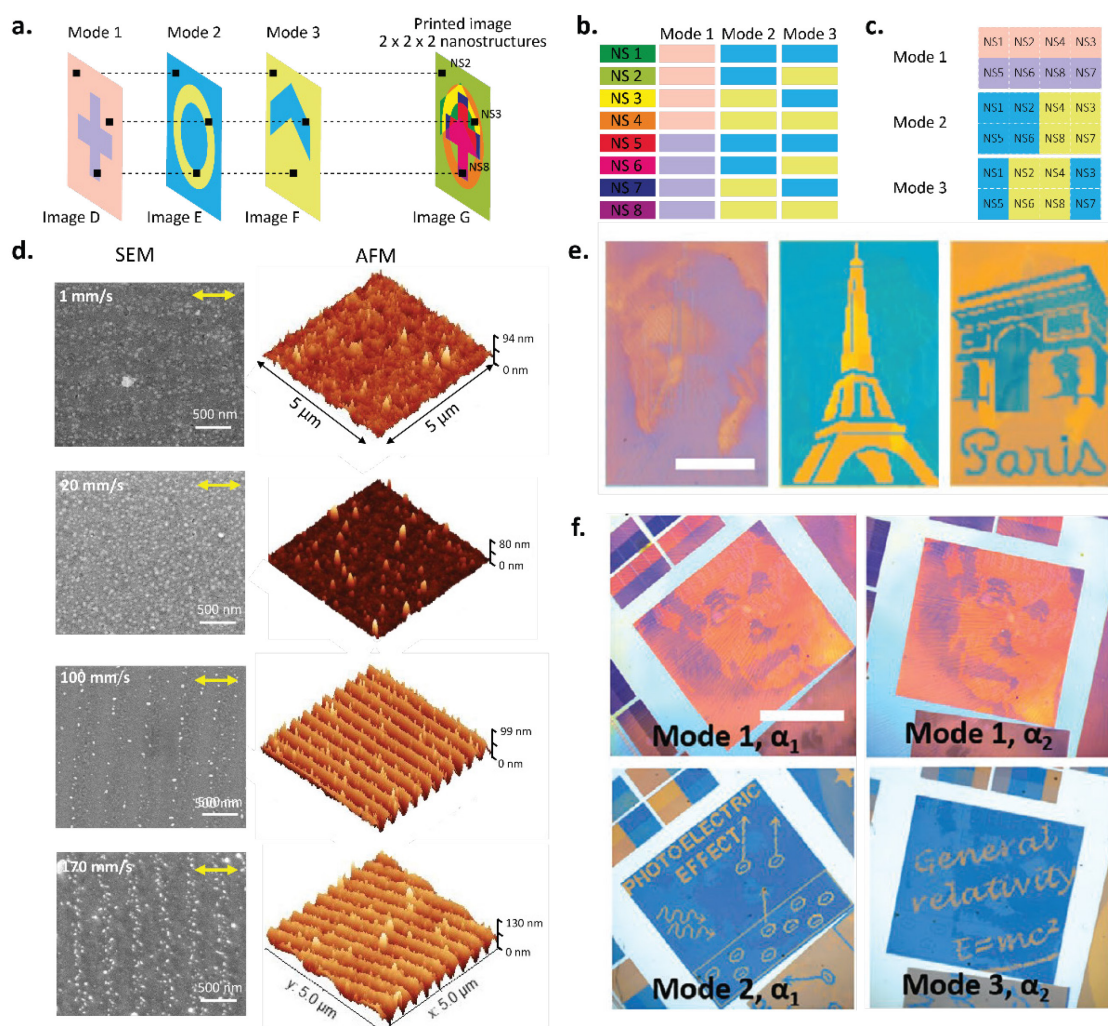


Figure 2.6 Printed image multiplexing realized by our group with femtosecond laser on Ag:TiO<sub>2</sub> nanocomposite films (a) Conditions for three image multiplexing. Three bicolor images are combined together to create a multiplexed image that can be printed by laser.

The latter displays independently one image in each mode provided that the nanostructured pixels (NS) used for printing exhibit the specific three-dimensional colors that are ordered in the logical trees (b). Here, the number of different types of nanostructured pixels, or of random plasmonic metasurfaces within the spatial extent of a pixel, required to print two bicolor multiplexed images is  $2 \times 2 \times 2 = 8$  for three bicolor images. Ordering these nanostructured pixels in a matrix form (c) offers a visual guide to easily identify whether all the required nanostructures for implementing multiplexing are present. (d) SEM and AFM images of the Ag:TiO<sub>2</sub> nanostructures produced by femtosecond laser irradiation at several scanning speeds. The embedded grating (EG) for speed  $v = 1 \text{ mm s}^{-1}$  is formed parallel to the laser polarization (depicted by double side yellow arrows), whereas the surface grating (SG) at  $v = 100 \text{ mm s}^{-1}$  and  $170 \text{ mm s}^{-1}$  is oriented perpendicular to the laser polarization. (e) Photographs of a first demonstrator displaying three images in modes reflection (Mode 1) and polarized transmission with two different angles of polarizers (Modes 2 and 3) (f) Photographs of a second demonstrator showing the change of displayed image consequently to a change of sample azimuth angle  $\alpha$  when placed between polarizers (Modes 2 and 3). (a,b,c,e,f) reproduced from [54], (d) reproduced from [75].

Using these shapes, we were able to identify when a color difference appeared between two squares which should have the same color. For a set of nanostructures that allow multiplexing, in each mode the nanostructures displaying the same color should be grouped in geometrical shapes (Figure 2.6c). With this combinatorial and visual approach to multiplexing, as well as by leveraging the polarization dichroism due to the anisotropy of our nanostructures, we were able to produce three-image multiplexing to be observed under in reflection and polarized transmission with two different angles of polarizers (Figure 2.6e,f). To the best of our knowledge, this is the first time three images are multiplexed under white light. Further, we are the first to precisely describe the way we select parameters for image multiplexing with plasmonic nanostructures. However, our visual and iterative exploration of the colors could hardly be generalized to other modes or more images, as it would require keeping track of the color variations in more than three modes at once. For this, an automated exploration would be much more convenient.

## 2.4 Conclusion

Image multiplexing, which consists in encoding two or more independent images in the same spot on a material, has a great potential in both anti-counterfeiting and information storage, and has thus been the subject of numerous studies. Among the different types of image multiplexing, *printed* image multiplexing, where the color of each pixel of the multiplexed image changes to display the encoded images in the different read-out modes, is the most promising for easy "level 1" authentication under white light with no additional device.

Such easily authenticable printed image multiplexing has been achieved with specific ink layouts on both paper and metallic substrates. For these, well-known analytical spectral prediction models allow the precise identification of ranges of ink amounts which allow multiplexing. However, due to the linear and smooth color variations obtained by ink-based printing, strong limitations apply on the achievable color combinations, resulting in at most two multiplexed images with limited colorfulness.

On the opposite, plasmonic nanostructures can display tremendous color variations thanks to strong local field enhancement. They provide an exciting new way for the ink-free printing of color images as well as printed image multiplexing. Precise nanofabrication techniques are implemented to create two-image multiplexing prints by fabricating various anisotropic nanostructures. With these, the design of the multiplexed print relies on the variation of controlled geometrical parameters, which relation to the color of each mode can be modeled with the help of computationally intensive simulations. However, the nanofabrication techniques required for the design of such nanostructures are expensive and often limited to small scale areas, which imposes the use of a microscope for authentication.

Several laser processes were proposed to overcome this shortcoming, providing high-speed large scale creation or modification of both plasmonic and dielectric nanostructures. These were used to implement the printed image multiplexing of two color images to be observed under polarized white light, which much faster processing speeds and lower costs. Such a laser process was successfully implemented by the Nanoparticles team at Laboratoire Hubert Curien, Saint-Etienne, using a femtosecond laser on Ag:TiO<sub>2</sub> nanocomposite films, resulting in the multiplexing of two images using reflection and polarized transmission, and of three images using reflection, and two different sample azimuth angles in polarized transmission by leveraging the anisotropy of the created metasurfaces.

Apart from our recent article, the number of modes in these works was always limited to two. In the case of ink-based printing, this might be understood due to the limited range of optical effects obtained with each technology. A combination of the proposed strategies, such as, for example, the printing of metallic inks on duplex prints could provide a higher number of modes, but the increased complexity of the ink layout might instead result in more limitations to the displayed images, and of course a more complex printing process.

In the case of plasmonic nanostructures, the potential for easily observable modes does not seem to have been fully explored yet. Although plasmonic nanostructures exhibit a wide range of optical effects visible in different modes, the only outcomes published so far require the use of polarizers, which are not always available. Besides, the color reproduction concepts inherited from ink-based printing, such as gamut mapping and halftoning, are rarely implemented with plasmonic printing, which lowers the final quality of the displayed images. We believe the strength of our recent work was to derive a first general framework for the identification of the nanostructures needed for multiplexing, which already allowed to use an additional mode, unpolarized reflection, which had not been identified in other works. To leverage a wider range of optical effects, as well as to produce higher quality multiplexed images with plasmonic nanostructures, there is a need for a highly versatile and automatic workflow to produce printed image multiplexing.

To conceive and implement this workflow, we decided to replace our femtosecond laser by a less costly nanosecond fiber laser, which longer pulses are expected to increase the processing speed due to an increase in thermal processes. As seen previously, laser processes, including nanosecond lasers, are capable of generating plasmonic nanostructures achieving a wide variety of optical effects. Compared to the more accurately defined structures obtained by top-down nanofabrication techniques, laser-induced nanostructures are created by a wide variety of phenomena, ranging from oxidation to melting and reshaping of the nanoparticles, including resonances supported by already present nanostructures. Therefore, their color is much more complex to



---

simulate and to relate to the processing parameters. However, these phenomena also have the advantage of modifying several morphological parameters of the metasurfaces at the same time, which results in much more variability in the produced nanostructures. To overcome the lack of predictive model and to leverage the multiple variations of metasurfaces available, we include the characterization of the colors produced by each laser parameters in our workflow as a first step towards the automatic identification of the nanostructures required for multiplexing. The next chapter describes this characterization and the resulting construction of metasurface color databases.

## Chapter 3.

# Production of metasurface color databases

Plasmonic metasurfaces consisting of Ag nanoparticles embedded in a TiO<sub>2</sub> mesoporous layer have shown to be promising for multiplexing due to the wide range of optical effects and colors they create. Using a femtosecond laser, our team already created such metasurfaces and used them to produce up to three-image multiplexing using polarizers. In this work, we use a nanosecond laser to create such metasurfaces with higher speeds and use their colors to print both color images and multiplexed images.

The first step towards this goal is to know which colors can be produced, in which mode their color should be observed, and which laser processing parameters produce them. However, there is not any model that can predict the created nanostructures, let alone their color, from the laser processing parameters. Therefore, all laser parameters have to be inscribed at least once, and the resulting metasurfaces' colors characterized, to be able to retrieve the produced colors from the set of laser parameters.

In the present chapter, we introduce the new nanosecond laser setup (Figure 3.1) that will be used throughout this thesis, as well as the method put in place to characterize the metasurfaces it produces when used to process Ag:TiO<sub>2</sub> nanocomposite samples. The first part of this chapter describes the laser setup, the way it is operated to produce metasurfaces so that they can subsequently be used in an image, and the nanoscale features of the produced nanostructures. This setup is used to produce metasurface databases by varying each laser parameter continuously, creating thousands of metasurfaces which colors has to be measured in several modes. Subsequently, the second part of this chapter describes the color measurement of these metasurfaces databases. We design a custom optical setup enabling both color and spectral measurements, to be able to measure the color of the metasurfaces in all the illumination and observation conditions required to capture their singular optical effects.

This setup is used to measure the color of each metasurface of the database in each mode to create a metasurface color database enabling us to link each set of laser parameters to the colors it produces in the considered modes. This metasurface color

database will subsequently be used as a look-up table to produce color images and multiplexed images in following chapters.

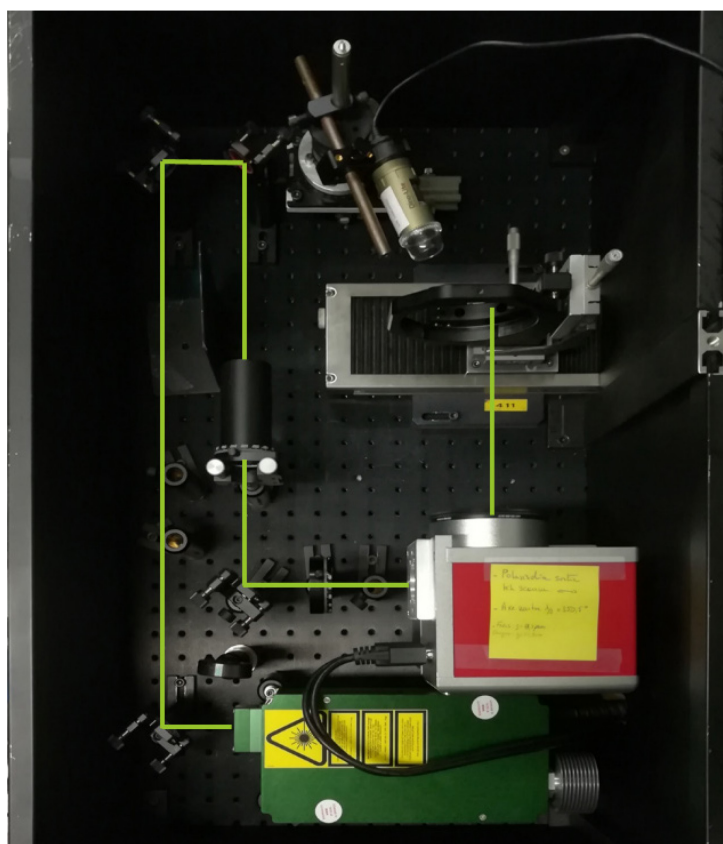


Figure 3.1 Nanosecond laser set-up The green line indicates the laser path.

### 3.1 Laser processing

In this part, we describe the nanosecond laser processing setup created for this work. The laser is operated to create metasurfaces on Ag:TiO<sub>2</sub> samples in a pixel-based manner. Two types of metasurface pixels are defined, which can subsequently be used inside images. A metasurface database is produced by varying continuously each laser processing parameters.

#### 3.1.1 Laser set-up

Our laser is an industrial fibered laser (*IPG Photonics*) of wavelength 532 nm, with a typical pulse duration of 1.3 ns. It is linearly polarized using a Brewster angle polarizer placed before the scanner head (see Figure 3.2 for an annotated sketch, and Figure 3.1 for the picture of the real setup). The polarization can be rotated using a half-wave plate. The laser is scanned over the sample with a galvanometric mirror scanner head (*Sunny Technology*). A scanning head consists of two galvanometric mirrors which are rotated in order to control the beam angle in both the horizontal and the vertical directions. The laser beam is then focused on the sample using a 16 cm F-Theta lens, which converts the

angle given by the galvanometric mirrors to a displacement in the writing field. This arrangement provides a maximum field size of  $11 \times 11 \text{ cm}^2$  in the focal plane of the lens. The focused laser spot size is  $13.5 \text{ }\mu\text{m}$  with a Rayleigh length of  $818 \text{ }\mu\text{m}$ .

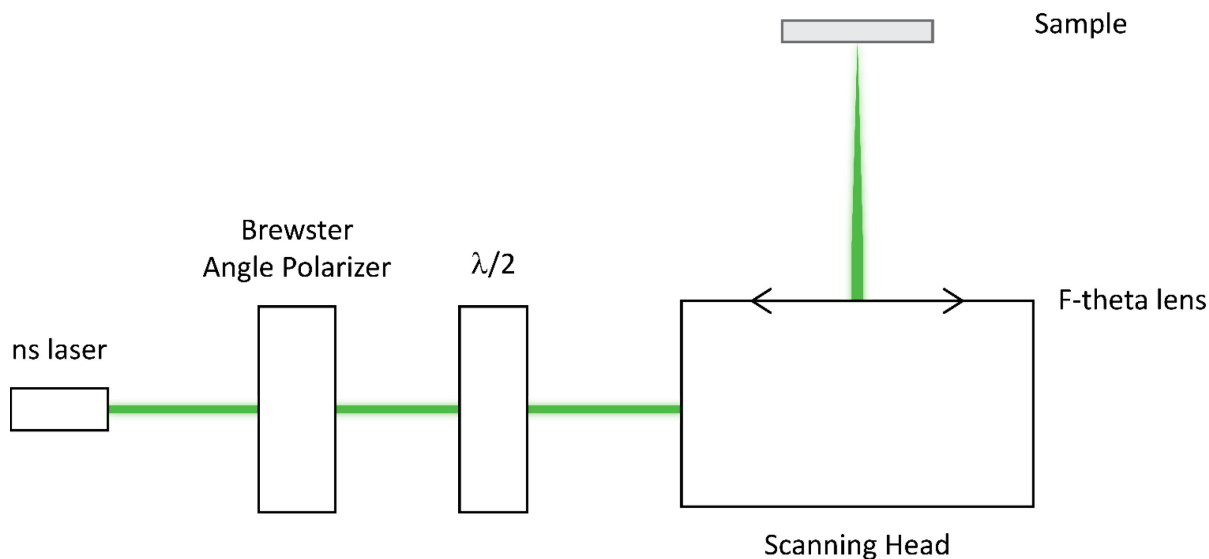


Figure 3.2 Sketch of the nanosecond laser processing setup.

### 3.1.2 Processing parameters

The laser and scanner head are controlled by the same software (*CS Mark*), which enables to control the scanning head to move the laser beam in the sample plane, as well as to tune the laser power and repetition rate during marking. The laser movements on the sample are controlled via a vectorial drawing interface. The software allows drawing dots, lines, and simple geometric shapes (rectangles, circles etc.) which define the path along which the laser will be moved. To each shape is attributed a set of laser processing parameters. These contain parameters related to the laser operation, namely power and repetition rate, as well as an additional shape-dependent parameter controlling the way the laser is scanned over the vector drawing:

- in the case of dots, this parameter is the *exposure time* during which the laser is turned on at the position of the dot, before shutting off and moving to the next shape.
- in the case of lines, this parameter is the *scanning speed* at which the laser is continuously scanned along the line.

Once the drawing and the laser and scanning parameters are defined, the software controls the way the scanning head and the laser are operated, following the vector drawing defined by the user, and automatically changing the laser power and speed. On the contrary, polarization is changed manually by rotating the half-wave plate, which requires having one separate vector drawing per desired laser polarization.

As the processes at the root of the metasurfaces creation with nanosecond pulses is not entirely known, it is assumed that all available laser processing parameters can have an effect on the created metasurface. They are therefore all varied throughout this thesis: **power, repetition rate, exposure time or scanning speed, and polarization.**

### 3.1.3 Laser processing elements

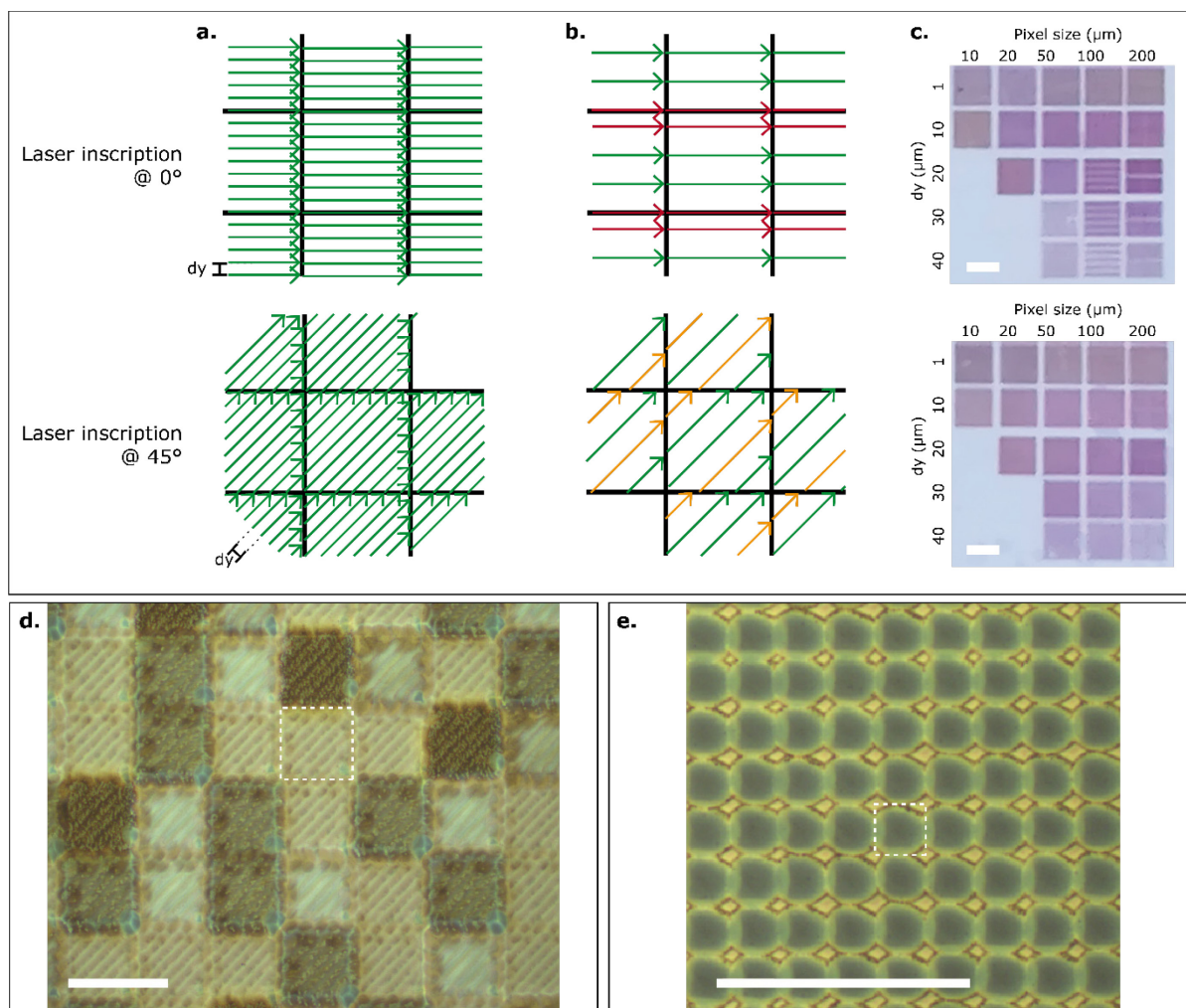
To be able to create an image with the metasurfaces, we have to define small elements which will be used as the pixels of the image. These elements are defined as vector drawings which can be imported in the laser processing software to control the laser processing. Two geometries of pixels have been used in this thesis, leveraging the two basis elements of the software: lines and dots. The first processing elements are “squares”, consisting of parallel lines that fill the square area. The second are “dots”, which can be directly defined in the software. Each geometry has its advantages and drawbacks and allows to vary additional geometry-specific parameters which will be described now.

#### *Squares*

Each pixel of the image is converted to a vectorized square, which is hatched with laser lines separated by interline spacing  $dy$  (Figure 3.3a). In a square pixel, laser lines can be drawn with an angle relative to the pixel side. Figure 3.3a shows two examples where this angle equals  $0^\circ$  or  $45^\circ$ . The  $dy$  distance can be set independently of the pixel size. However, when  $dy$  is not proportional to the pixel size in the direction of the hatching, the lines closest to the pixel border have a different interline distance than the others (Figure 3.3b). This can result in a change of the average nanostructure colors, and to the presence of a moiré phenomenon in the pixel borders when looking at the sample with an appropriate resolution (Figure 3.3c). With  $0^\circ$  laser lines orientation, the  $dy$  error happens at each horizontal pixel border whereas with  $45^\circ$  laser lines orientation, the misaligned lines only affect the filling factor in the square corner, which has much less impact on the average color of the pixel. Therefore, an orientation of  $45^\circ$  was chosen in this article. Figure 3.3d shows a photograph of resulting laser processed pixels taken with an optical microscope.

#### *Dots*

Each pixel of the image is converted to a single vectorized dot. The spacing between the dots was set to  $20\ \mu\text{m}$  (Figure 3.3e). The colored area within the  $20^2\ \mu\text{m}^2$  square depends on the laser parameters used (laser fluence, polarization, repetition rate and exposure time). The accumulated number of pulses in a dot equals the product of the repetition rate and the exposure time.



*Figure 3.3 Laser processed pixels using lines or dots. From a) to d), each pixel is printed by drawing laser lines separated by the interline spacing  $dy$ , either at  $0^\circ$  relative to the square edge or at  $45^\circ$ . (a) Both orientations are suitable if the pixel size in the normal direction is proportional to  $dy$ . (b) If the pixel size is not proportional to  $dy$ , the  $0^\circ$  laser orientation creates moiré lines that are clearly visible in images (c top) recorded with the setup presented in Figure 3.8. The  $45^\circ$  line orientation creates less visible moiré effects (c bottom). The length of the scale bars is  $500 \mu\text{m}$  in images c). (d) Optical microscope image of  $72^2 \mu\text{m}^2$  square pixels filled with lines at  $45^\circ$  with  $dy = 10 \mu\text{m}$ . (e) Optical microscope image of  $20^2 \mu\text{m}^2$  dot pixels. In (d) and (e), the white dashed squares highlight the area of one pixel and the length of the scale bar is  $100 \mu\text{m}$ .*

Note that the square pixels introduce a new parameter than can be varied: the **interline spacing**, which makes five independent parameters to be varied for this pixel geometry, and four for dot pixels.

Due to the gaussian shape of the beam, both laser processing geometries create borders in which the deposited laser energy can vary strongly from the center of the pixel. In the following section, we show the nanostructures created at the center of pixels, which are the main driver of the optical effects created. However, due to this border effect, the

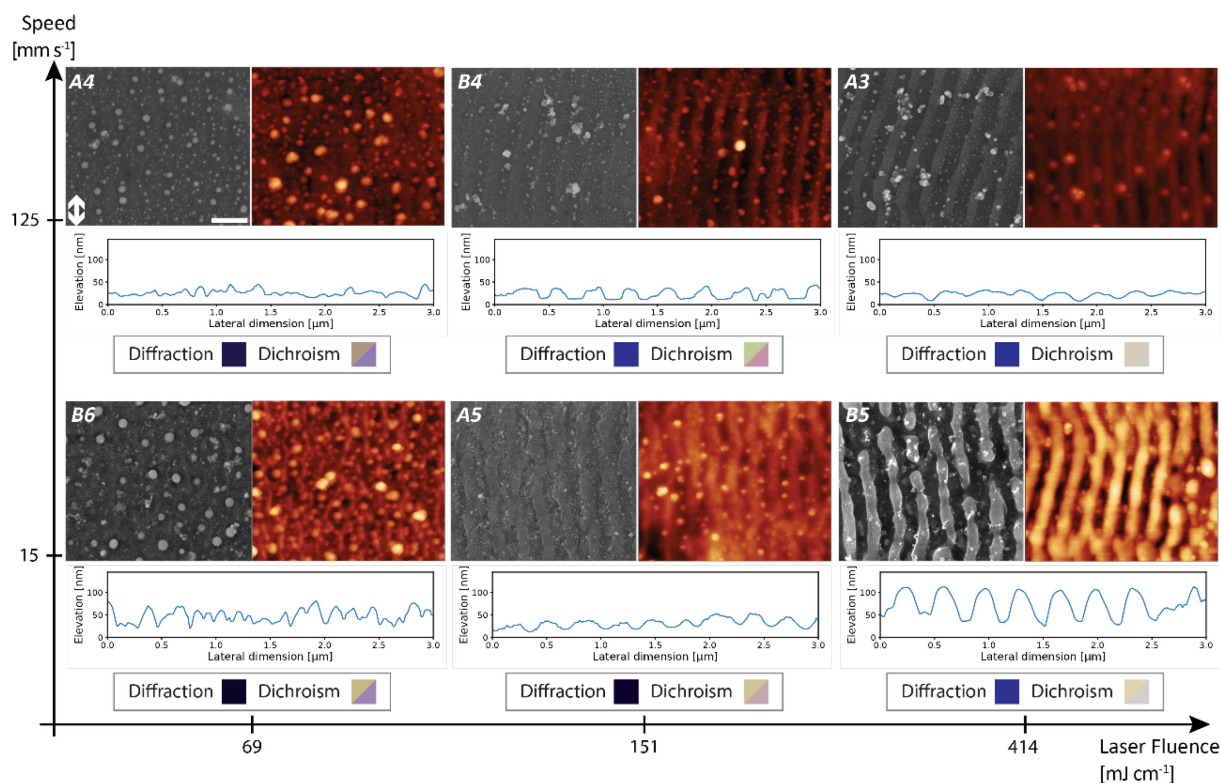
color that will be considered for one metasurface in the second part of this chapter is an average of several identical pixels.

### 3.1.4 Formed metasurfaces

The results reported below are the first ones demonstrating nanosecond-laser-induced self-organization of silver nanoparticles in TiO<sub>2</sub> thin films. The longer pulse duration (1.3 ns) strongly alters the mechanisms leading to the nanoparticle growth. As expected with higher thermal effects, the number of pulses required to grow silver nanoparticles and get various colors is smaller than what is required with a femtosecond laser at similar repetition rate. Compared to the 10 mm/s scan speed used to print multiplexed images with a femtosecond laser (190 fs pulse duration) at 515 nm wavelength and 500 kHz repetition rate [54], the metasurfaces presented in this thesis involve scan speeds up to 2,000 mm/s. Thus, implementing nanosecond laser processing reduces the printing time in average by about a factor of 50.

A selection of characteristic self-organized structures exhibiting dichroism, diffraction or both simultaneously, is presented in Figure 3.4. This selection gives the reader an overview of the nanostructure properties at the origin of the singular colors required for multiplexing. At low fluence (69 mJ cm<sup>-2</sup>), Ag nanoparticles tend to grow near the top surface and align roughly along the laser polarization, making the metasurface dichroic for both speeds, as highlighted by the different colors recorded in the two polarized modes. When comparing the 15 and 125 mm s<sup>-1</sup> speeds, it appears that bigger nanoparticles are grown at lower speed even though a large density of small Ag nanoparticles still fills the metasurface. At the fluence of 151 mJ cm<sup>-2</sup>, self-organized gratings are formed on the film surface and the Ag nanoparticle size remains small in overall. A modulation of the nanoparticle density between the grating grooves and ridges characterizes the metasurface at high speed at this fluence. At 414 mJ cm<sup>-2</sup>, the grating grooves are deeper and the Ag nanoparticle density is lower. A stronger topographic contrast appears at low speed in this case. The period of the self-organized gratings lies in the range between 290 nm and 340 nm. The grating lines are parallel to the incident laser polarization. Interestingly, the anisotropy of the nanogratings with the highest contrast at higher fluence is negligible, meaning that the optical anisotropy is more related to the presence of large nanoparticles or a high density of small nanoparticles preferentially aligned in one direction than to the presence of surface grating-like structures. Conversely, the diffraction efficiency is much higher for the surface grating-like structures obtained at high fluence than for metasurfaces with bigger nanoparticles obtained at low fluence. At intermediate fluences, the modulation of the nanoparticle density observed at high speed also diffracts light. As the period value is lower than the incident wavelengths (white light), the -1<sup>st</sup> diffracted order only appears for an incidence angle larger than 10° for the blue (400 nm) part of the spectrum and is measured in the backscattering configuration described in Figure 3.7. When comparing these

metasurfaces with the ones produced by femtosecond lasers[13], [54], [75], [77], the embedded nanoparticles can have higher sizes at low speed and the surface grating-like structures are parallel to the laser polarization while they were perpendicular to the laser polarization with the femtosecond laser processing.



*Figure 3.4. Nanostructured metasurfaces produced by ns laser processing. SEM and AFM micrographs and topographic profile perpendicular to laser polarization, of six of the twelve metasurfaces subsequently shown in Figure 3.7, obtained at various scan speeds and laser fluences. At this repetition rate (300 kHz) and with the  $1/e$  laser spot size of  $13.5 \mu\text{m}$ , the speeds  $15 \text{ mm s}^{-1}$  and  $125 \text{ mm s}^{-1}$  correspond to an effective number of pulses of 1 080 and 129.6, respectively. For each structure, the color of the inset square indicates the presence of diffraction (blue) and the polarization-dependent dichroism (difference of colors between the two corners of the inset square). The scale bar is the same for all SEM and AFM images and corresponds to  $500 \text{ nm}$ . The double-sided white arrow gives the direction of the laser polarization during processing. The laser scanning direction for printing the squares is at  $45^\circ$  from the double arrow.*

The presence of metallic nanoparticles in the thin  $\text{TiO}_2$  film and subwavelength gratings is at the origin of various optical phenomena, such as localized surface plasmon resonances, thin film interferences, diffraction, and hybridization of resonances. A deep understanding of these mechanisms is beyond the scope of this work, which demonstrates a more empirical approach applicable to any color printing technique enabling observation in different modes. Additional characterization and simulations on these metasurfaces have been carried out by other members of the team and was published in a recent article[78].

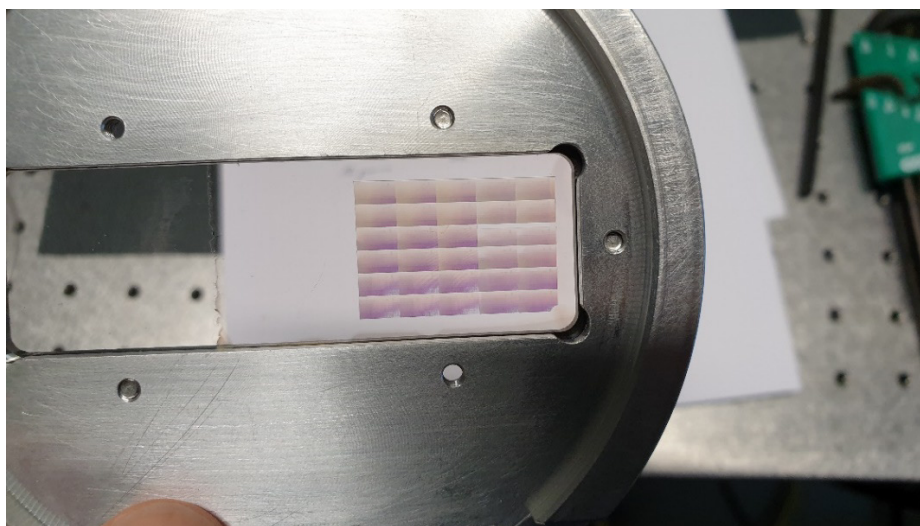


### 3.1.5 Metasurface database

We create a metasurface database by sampling each laser parameters to make the color vary as regularly as possible in all modes, based on both knowledge from femto-second and continuous laser experiments, and on visual trial and error. With the absence of a model predicting the color of the metasurfaces from the laser parameters, and given the number of parameters and modes, designing the database based on our empirical knowledge is the most practical approach to obtain interesting colors while reducing the space require to print all sets of laser parameters. Indeed, the high number of laser parameters to be printed would require a vast space if they were all sampled linearly. Moreover, considering that some color changes happen very rapidly, this would put us at risk of missing some colors. On the other hand, a trained observer identifies easily the parameters ranges in which the sampling has to be increased and those in which the sampling can be decreased.

Another possibility would have been to adapt the procedure described in [65], which implements a multi-objective optimization scheme to achieve the best sampling to obtain a wide range of colors in one mode. However, the high dimensionality of the performance space in our case (color in seven modes, thickness of the pixel, possible multiplexing solutions) renders this task difficult.

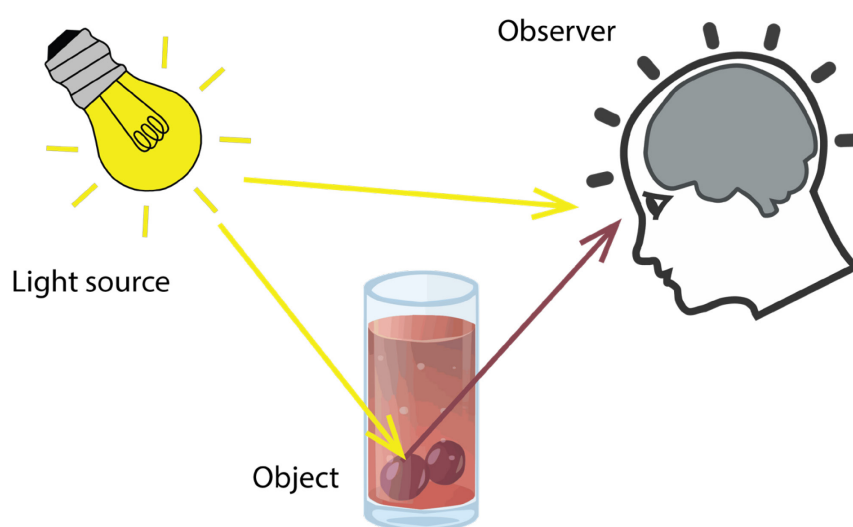
Figure 3.5 shows a glass sampled covered with the Ag:TiO<sub>2</sub> film on which part of the database has been inscribed.



*Figure 3.5 7560 metasurfaces produced within  $216^2 \mu\text{m}^2$  squares by varying 5 laser parameters. Each square is made of 9 pixels of  $72^2 \mu\text{m}^2$  area printed with lines at  $45^\circ$ .*

## 3.2 Multi-mode color measurement

Object color is a feeling created by the interaction of three components : the light source, the object, and the observer (Figure 3.6). It is thus traditionally described by three spectral quantities : the light source, the reflectance or transmittance of the object, and the sensitivities of the light receptors of the observer. However, in our case, the object has several colors depending on the modes. Instead of color, we introduce the term “hypercolor” which describes the set of colors produced by a metasurface when it is observed in different modes. To characterize these metasurface hypercolors, we have to define the modes precisely and measure all metasurfaces in fixed and repeatable conditions.



*Figure 3.6 Color perception originates from the interactions between the light source, the object, and the observer.*

Measuring the reflectance or transmittance of the object is usually done using a spectrophotometer. They operate in well-defined and standard observation conditions and seldom incorporate two polarizers. However, in our case, the color of the metasurfaces needs to be measured in a wide variety of illumination and observation conditions, to be able to identify the modes that will be used for multiplexing. As seen in the previous section, given the high number of parameters that can be varied (up to five) in the production of the metasurfaces, and their high and non-linear effect on color, the number of nanostructured patches to be measured to obtain a good sampling of the parameter space is high. Thus, the area of the patches needs to be very small with respect to the inscription field, which in turn requires the measurement of smaller areas than what is usually possible with spectrophotometers. This led us to consider an imaging setup with a color camera and automated rotations to be able to characterize the nanostructured samples in all the interesting modes with a high repeatability and a magnification allowing us to resolve our marked images up to a single pixel. All

characterizations and observations are done in this controlled set-up where the light sources and the characteristics of the observer are fixed. To define these modes properly, we first take a look at a few representative metasurfaces and their optical effects.

### 1.1.1 Optical properties of the metasurfaces

We pay attention to a selection of 12 metasurfaces (whose laser processing parameters are given in Appendix A, Table A.1) to present the interesting optical properties these random plasmonic metasurfaces exhibit. Figure 3.7 depicts the seven observation modes used in this work and shows the colors measured in each mode for the 12 metasurfaces under incoherent white light illumination. The color of each metasurface depends on the observation mode. Five modes with non-polarized light are considered. They correspond to “front side reflection” (Figure 3.7a), “backside reflection” (Figure 3.7b), “scattering” (Figure 3.7c), “diffraction” (Figure 3.7d) and “transmission” (Figure 3.7e) configurations. In addition, two modes correspond to a configuration where the sample is observed in transmission between two polarizers crossed at  $85^\circ$  from each other, the first one being vertical. These two modes, named “polarized transmission 1” (Figure 3.7f) and “polarized transmission 2” (Figure 3.7g) correspond to two azimuthal angles of the sample,  $\alpha$  equals  $45^\circ$  or  $135^\circ$ , relative to the direction of the vertical polarizer.

One can note that the color palette accessible with these metasurfaces in “backside reflection” is very broad, whereas “frontside reflection” and “unpolarized transmission” exhibit more shades of similar hues. However, the lightness contrast in these modes can be stronger. For example, there is a strong contrast between A4 and B1 or B5 in “frontside reflection”, or between A4 and B1 in “unpolarized transmission”. Some metasurfaces, like A3, are diffractive and light up in the “diffraction” mode (Figure 3.7d), while other metasurfaces, like B3, which has the same color as A3 in the three transmission modes, do not diffract and remain black in the “diffraction” mode. Some metasurfaces also exhibit dichroic properties, i.e. a sensitivity of their complex refractive index to light polarization, as highlighted by squares A6 and B6, showing different colors in “polarized transmission 1” (green) and “polarized transmission 2” (purple) modes. Interestingly, while some metasurfaces are both diffractive and dichroic (B4), others are diffractive and non-dichroic (A3), or non-diffractive but dichroic (B6) or even neither diffractive nor dichroic (B1).

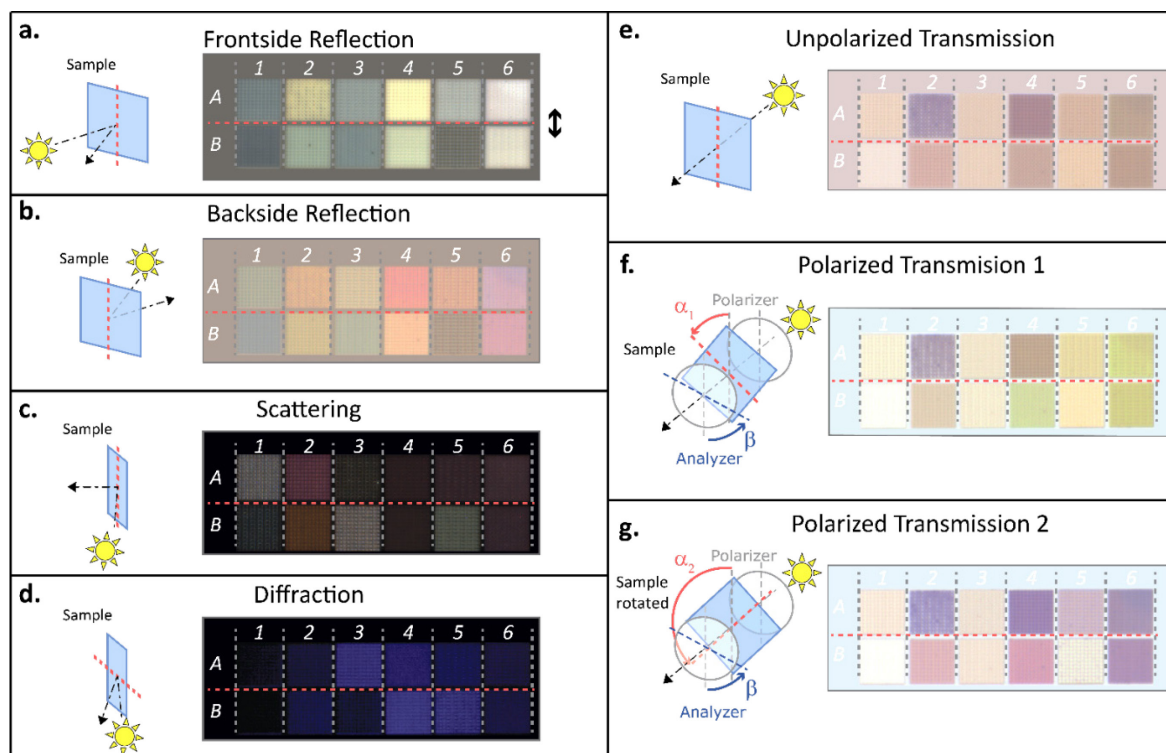


Figure 3.7 Colors of 12 laser-processed squares recorded in 7 observation modes. The laser processing parameters used to print the 12 squares are given in Table A.1. The laser polarization (double-sided black arrow shown in a) is parallel to the vertical side of all squares (laser polarization oriented at  $0^\circ$ ). The orientation of the horizontal side of squares is indicated by a dashed red line in all sketches drawn on the left-end side of the colored images. For each observation mode, the sketch illustrates the optical setup used to record the color images. Measurements were carried out with a mean incidence angle of  $0^\circ$  in transmission and of  $16^\circ$  in reflection. In the scattering mode, the camera axis is normal to the sample surface and the incidence angle is  $32^\circ$ . Diffraction catches the -1st diffraction order for an incidence angle of  $64^\circ$ . For the polarized transmission modes, one polarizer is placed between the source and the sample, and another one between the sample and the camera. The polarizers can be rotated independently in their plane. In the sketches of the polarized modes f and g, the azimuthal angle  $\alpha$  of the sample corresponds to the angle of the horizontal side of a square relative to the polarizer axis. It is set to  $45^\circ$  in f and to  $135^\circ$  in g. Angle  $\beta$  corresponds to the angle of the analyzer axis relative to the polarizer axis, which is set to  $85^\circ$  in f and g. The side of each square is 0.864 mm long.

The diffraction is due to the presence of laser-induced self-organized gratings[13]. The low period ( $< 400$  nm) of such gratings makes them highly dispersive for visible light, and hence explains why a single color (blue for the chosen angles in the “diffraction” mode) reaches the camera sensor. In Figure 3.7, all the gratings present in squares A3, A4, B4, and B5 are perpendicular to the incidence plane, in which the detector is placed. When rotating the gratings in their plane, conical diffraction occurs, and a detector placed in the plane of incidence cannot catch diffraction anymore. A diffractive grating whose orientation is not perpendicular to the incidence plane can thus appear black in the

diffraction mode [77]. The grating orientation is then used as an encoding parameter for multiplexed diffractive images later in this thesis.

To be able to measure the color of each metasurface in the database, in each of the discussed modes, we designed a custom optical setup.

### 3.2.1 Color measurement setup

Our color measurement setup is constructed as follows (Figure 3.8). A telecentric lens images the sample surface on a RGB camera (*Basler ac2500-14gc*). Two 15 cm wide LED light source are positioned in two ways, “Transmission” or “Reflection” to accommodate the various geometries of the illumination-observation modes considered in this study, used one at a time. The sample can rotate around an axis perpendicular to the plane of incidence to change the incidence and observation angles (the source being broad, the incident beam on each point of the sample has a wide angular aperture) and in its plane to change the azimuthal angle. In the transmission mode, the average incidence angle on the sample is null. In the reflection mode, the average incidence angle is  $16^\circ$ . In the diffraction mode, the source and the camera are kept in the same configuration as for the reflection mode, only the incidence angle is set to  $64^\circ$  to catch the -1st diffraction order in a backscattering configuration on the camera. In the scattering mode, the source and the camera are kept in the same configuration as for the reflection mode, only the incidence angle is set to  $32^\circ$ , meaning that the observation is normal to the sample. A polarizer and an analyzer can be inserted along the light path before and after the sample in the “Transmission” mode to enable Polarized Transmission modes. Two rotations, one on the sample azimuth (sample rotating in its plane) and one on the analyzer allows to set the angles  $\alpha$  and  $\beta$  defined in Figure 3.7f.  $\beta$ , the angle between the two polarizers, is arbitrarily set at  $85^\circ$ , to underline the sample dichroism.  $\alpha$ , which corresponds to the angle between the horizontal side of a square and the polarizer axis is set at  $45^\circ$  or at  $135^\circ$  to get the larger color difference between the two polarized transmission modes when a square is dichroic. In the “frontside reflection” mode, the incident medium on the metasurface is the air, while it is the glass substrate in the backside reflection mode. In the diffraction and scattering modes, the sample is illuminated from the backside. Results are the same in transmission when the sample is illuminated from front side or backside.

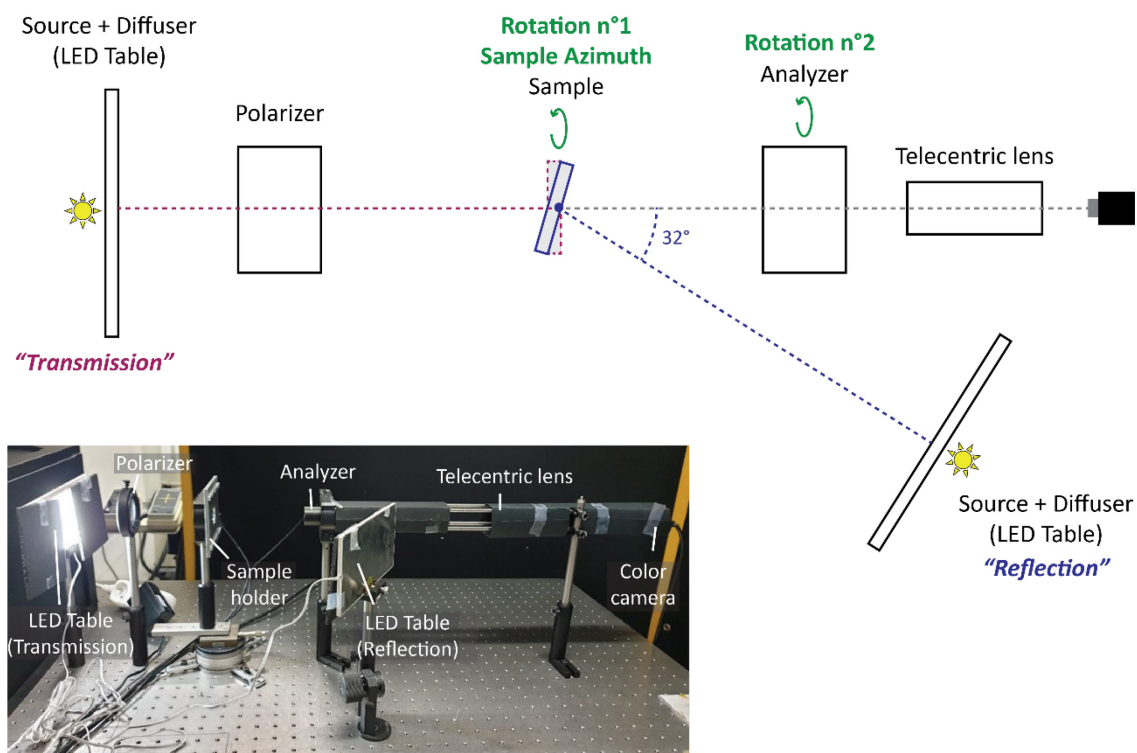


Figure 3.8 Custom color image acquisition set-up. Sketch and picture showing its operation in transmission mode.

### 3.2.2 Telecentric lens properties

Our telecentric lens is made with a simple 2 lenses set-up with focal lengths  $F = 250$  mm and  $F = 100$  separated by an aperture (diameter 0.8 mm) placed at the focal plane of both lenses. It is thus telecentric in both the object and the image spaces. Two main properties of the telecentric lens are of interest in the case of the characterization of our Ag:TiO<sub>2</sub> metasurface.

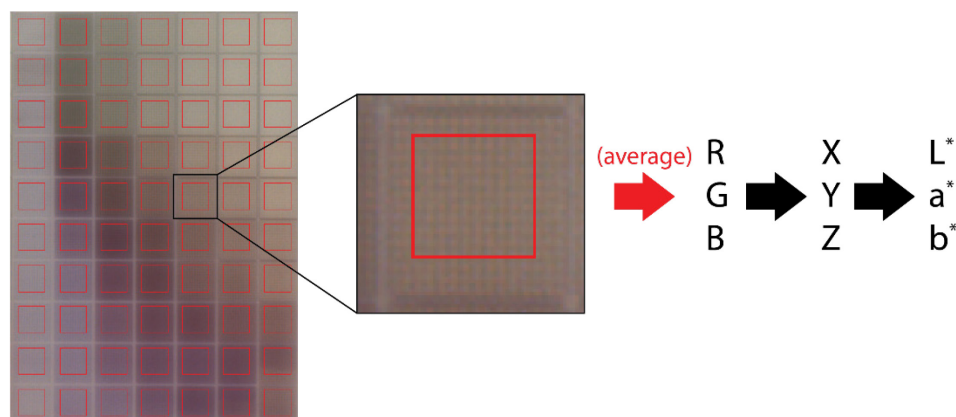
First, the average direction of the rays coming from the object that are imaged on the sensor is parallel to the optical axis in the object space, for every point of the image field. This allows to define precisely the direction of observation, and is particularly helpful in modes in which the color depends strongly on the incidence or observation angles (e.g. mode “scattering”).

Second, the object is orthographically projected onto the sensor, which means the dimensions of the object in its plane are not dependent on its distance to the lens. This has the effect of preserving the shape of the object when observing it at an angle, which greatly facilitates the subsequent image processing needed to extract the colors.

### 3.2.3 Image processing and color extraction

To measure the color of the metasurfaces, in the metasurface database, several laser-processed pixels are produced with the same parameters and arranged in squares. A picture is taken with our setup and normalized (Appendix B). As mentioned previously,

the color at the center of each pixel might be different from the color of its sides. Therefore, the camera pixel values are averaged over a large area covering several inscribed pixels of the same metasurface (Figure 3.9, red squares) before extraction.



*Figure 3.9 Color extraction procedure for one metasurface. Each metasurface is produced in several laser processed pixels arranged in squares (here, one square contains 15 x 15 laser processed pixels). The color is averaged on several pixels around the center of the square (red squares). In this picture, the dimensions of the squares in image pixels (on the camera) are 60 x 60. The size of the red squares in image pixels is 40 x 40, which results in the color being averaged on approximately 10 x 10 laser processed pixels. The extracted average RGB values for each square are transformed to XYZ and L\*a\*b\* according to the procedure described hereinafter.*

The conversion of the RGB values provided by the camera into colorimetric values that make sense perceptually need some comments. On the one side, the RGB values measured are device-dependent: they depend upon the spectral power distribution of the light source that we use, and the spectral response of the camera in the red, green and blue channels. On the other side, the colorimetric values according to standards proposed by the International Commission for Illumination (CIE) take into account chromatic adaptation, a perceptual phenomenon thanks to which the brightest light signal viewed (the *white point*) is systematically interpreted as white, the other colors being transformed accordingly and perceived, to some extent, independently of the color of the lighting. Hence, whatever the spectral response of the camera (or of the observer's sensors) is, the white color will be white, and black color will be black. The other colors might be slightly affected by the sensor responses, their colorimetric values are uncertain in absolute. But we argue that their ordering and their respective distances from each other is almost not affected, and the error that we make for their absolute value is anyway small compared to the error that we make by using the standard color assessment methods, developed for rather matte and opaque samples, with our specular/transparent, iridescent and goniochromatic samples. These standard color assessment methods are however the best options that we have in absence of existing specific color appearance models for our type of samples, while developing one would

need a long and tedious development based on many psycho-physical experiments that were not the aim of our project. Finally, we adopted the following summarized below and detailed in Appendix C.

The RGB values captured by the camera under our specific lighting are assumed to be sRGB values, a standard for RGB values following the CIE 61966-2-1 standard, assuming that the white point corresponds to the illuminant D65 reproducing the color of daylight. They are transformed by a standard linear transform into CIE 1931 XYZ tristimulus values, a standard representation for the color of light signals, by again considering the D65 illuminant as white point. Then, the XYZ tristimulus values are converted in CIE 1976 L\*a\*b\* coordinates. The CIE 1976 L\*a\*b\* color space is a basic color appearance model widely used in color science, especially in the printing industry, that is considered (perceptually) *uniform* and allows comparison between colors thanks to a perceptual color distance metric. As it is a color appearance model, it takes into account the chromatic adaptation (although in a rather basic way) by dividing the  $X$ ,  $Y$  and  $Z$  values of each sample with the ones,  $X_w$ ,  $Y_w$  and  $Z_w$  of the white point. As said above, this is thanks to this chromatic adaptation that we can consider that the colorimetric values are almost independent of the setup we used and prevented a tedious color calibration of the camera in each illumination-observation mode.

The main issue is the identification of the reference sample that forms the white point in each mode. We need to define, for each mode, an achromatic sample which reflects, or transmits similar amount of light towards the observer as the metasurfaces do, or even a little more to be sure that it will be the brightest object in the visual field and therefore interpreted by the visual system as a white object. In practice, such a sample does not yet exist and would be hard to produce physically. We therefore propose to proceed in a similar manner as in Ref. [79] for fluorescent inks, by considering a pseudo white defined with a luminance higher than all the luminances of our metasurfaces in the considered mode, having the same chromaticity as the D65 illuminant. Let  $x_{D65}$ ,  $y_{D65}$  be the chromaticities of the D65 illuminant, and  $Y_{D65} = 100$  its luminance, and  $Y_{max}$  the maximum luminance of the metasurfaces. The tristimulus values  $X_W Y_W Z_W$  of the pseudo white are computed as follows:

$$\begin{cases} X_W = x_{D65} \times Y_{max} / 75 \\ Y_W = y_{D65} \times Y_{max} / 75 \\ Z_W = z_{D65} \times Y_{max} / 75 \end{cases} \quad (3.1)$$

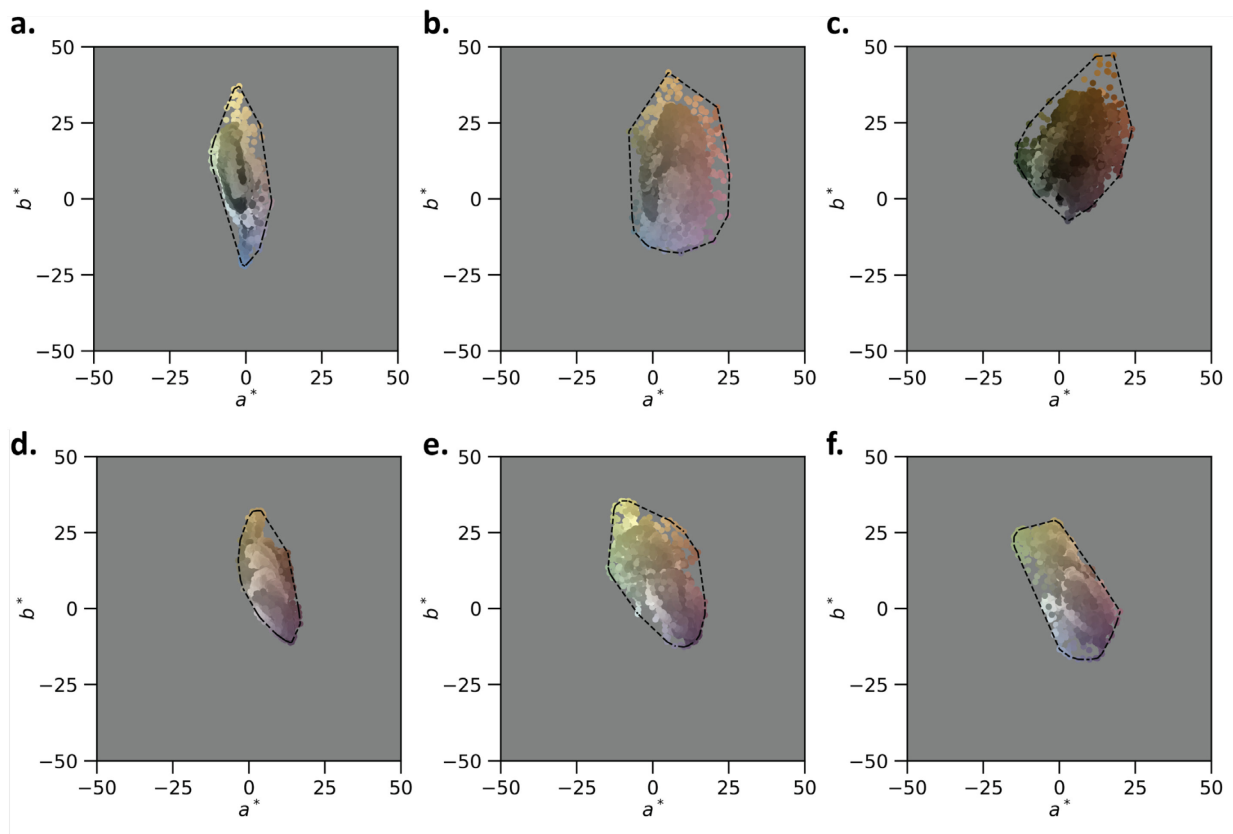
Then, the CIE 1976 L\*a\*b\* color coordinates can be computed for the considered mode, according to the formulas given in Appendix C. This procedure with the pseudo white ensures that the maximum  $L^*$  value in each mode will be close to 90 ( $L_{max}^* = 116 * (0.75)^{1/3} - 16 = 89.4$ ). This provides a homogeneous procedure to compute the color of our metasurfaces from the captured RGB images.



### 3.2.4 Color gamuts

By measuring our metasurfaces database in each mode with this fixed set-up, a metasurface hypercolor database is constructed. This allows us to know which color can be produced by our nanosecond laser process in each of the modes. The range of attainable color with a printing process is usually referred to as the gamut. As the color of the metasurfaces depend on the mode in which they are observed, we obtain several gamut, one for each mode.

Figure 3.10 shows the gamuts obtained with both pixel geometries (squares and dots) in each mode. As expected from the observation of the twelve metasurfaces in Figure 3.7, the widest color gamuts are obtained in “backside reflection”, “scattering” and “polarized transmission 1 & 2”, whereas the strongest lightness contrasts are obtained in “frontside reflection” and “unpolarized transmission”.

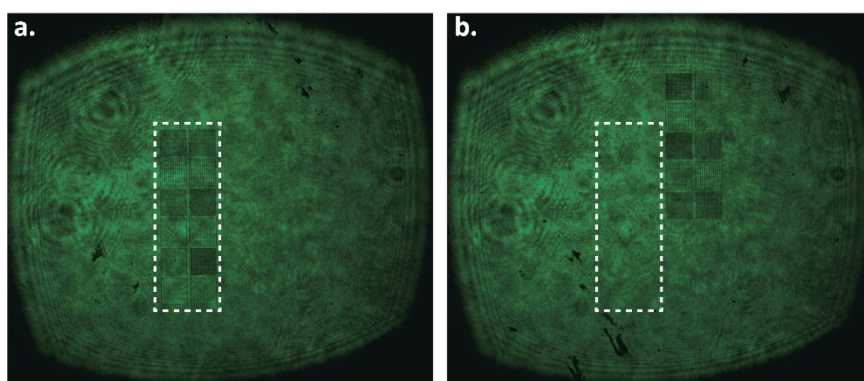


*Figure 3.10 Color gamuts in CIE  $a^*b^*$  plane from our metasurface database, characterized in 7 observation modes. (a) mode “frontside reflection” (b) mode “backside reflection” (c) mode “scattering” (d) mode “unpolarized transmission” (e) mode “polarized transmission 1” (f) mode “polarized transmission 2”. Each dot is colored with the sRGB color of the corresponding metasurface. No gamut is shown in diffraction as color is angle-dependent and only the brightness variation is used in this mode.*

This hypercolor database allows us to explore the available colors using our metasurfaces, without any predictive model. The following chapter describes how these colors can be used to produce an image.

### 3.2.5 Spectral measurements

This setup can be modified to measure the transmission and reflection spectra of the metasurfaces, by replacing the white light sources with a monochromatic light which wavelength can be tuned across the visual spectrum. By taking one picture for every wavelength, we are able to reconstruct a hyperspectral image, where one spectrum is measured for each pixel of the image. An accordable light source is constructed with a supercontinuum laser (New-wave, *Leukos*) and a wavelength-tunable filter (BEBOP, *Leukos*). The output is a collimated beam which is expanded and directed to the sample with mirrors. Two optical paths are created, one for transmission, one for reflection.



*Figure 3.11 Color images taken with monochromatic illumination at 550 nm (a) Image where the considered 12 metasurfaces are placed in the area of interest (white dotted line) (b) Reference image where the sample is rotated to place only the unprocessed film in the area of interest.*

At each wavelength, two color images are captured: one with the considered metasurfaces, and a reference one with the unprocessed film only. They are converted to grayscale by summing the three R, G, B, channels. The image of the metasurfaces is divided by the reference image, which gives us a value for transmission (respectively reflection) relative to the unprocessed film in each pixel at each wavelength. Doing so in a pixel by pixel manner allows us to correct the interference patterns caused by the coherence of the source (Figure 3.11). To reduce noise, the transmission value for each metasurfaces is obtained by taking the average of several pixels inside the corresponding square, similar to what is done for color measurement. The values for the different wavelengths are concatenated and the resulting spectra are multiplied by the transmittance (resp. reflectance) of the film to retrieve the absolute transmittance (resp. reflectance) in each pixel.

Although the spectral information is usually preferred, the hyperspectral measurement process is cumbersome and cannot be performed in some of the modes due to sensitivity limitations (e.g. scattering). Only the color measurement is used in the following.

### 3.3 Conclusion

Nanosecond laser processing of  $\text{TiO}_2\text{:Ag}$  nanocomposites enables producing a wide variety of visual effects, ranging from diffraction to polarization-sensitive dichroism, including bright specular reflection and transmission colors. The metasurfaces produced are thus promising for printed image multiplexing, offering a wide variety of modes. We identified seven of these modes in which the color of the metasurfaces should be characterized, and produced thousands of metasurfaces to construct a metasurface color database.

Our implementation of nanosecond laser processing with a galvanometric scanning head allows us to inscribe metasurfaces with a high scanning speed, up to  $2000 \text{ mm s}^{-1}$ . We show that the reshaping and self-organization of Ag nanoparticles in this  $\text{TiO}_2$  matrix happen at speeds much higher than with femtosecond and continuous lasers, which is highly valuable for further industrialization of the process. Nanosecond laser processing is thus a versatile and low cost processing technology which has a great potential for the laser marking of color images and multiplexed prints, due to its capabilities to process large areas with a high speed.

To counter the lack of a model predicting the colors of the metasurfaces from the laser processing parameters used to create them, we put in place a complete characterization scheme for multi-mode color acquisition using RGB imaging, which allows the creation of large hypercolor databases in multiple non-conventional modes of illumination and observation. The telecentric lens allows controlling precisely the angles measured by the setup and allows easy color extraction with its orthographic projection. Although spectral measurements can be performed with this setup by adding an accordable monochromatic light source, direct color measurement with the color camera and white light sources showed good results and is preferred due to its simplicity and ability to measure a large number of metasurfaces at once.

Our laser processing and multi-mode characterization scheme allows the controlled production of  $\text{TiO}_2\text{:Ag}$  metasurfaces at a pixel scale, by defining small laser processing elements than can be used as pixels of the processed image. This allows to use the metasurface colors in an image, and opens the way to subsequent image multiplexing. However, to reproduce any arbitrary color image, the right metasurface has to be automatically selected for each of the pixels of the desired image. Further, the range of color available is highly dependent on the considered mode, as showed by the color gamut measured and displayed in this chapter. To be printed correctly, the images have

---

to be adapted in order to be reproducible with our laser processing. The next chapter proposes a way to adapt and reproduce any color image using the metasurfaces of our database. To do this, concepts from the traditional ink-based printing techniques are adapted to laser processing to derive a simple yet effective color reproduction workflow.

## Chapter 4.

# Production of images

The relationship between laser processing parameters, metasurfaces, and their color in each mode has been established in the previous chapter by producing and characterizing metasurface color databases made of thousands of different nanostructures created with our nanosecond laser set-up. Owing to their specific optical properties, Ag:TiO<sub>2</sub> metasurfaces display very different colors depending on the considered modes. As shown in the previous chapter, this means that the gamut obtained with this laser processing method is highly dependent on the considered mode.

As with other printing techniques, the color of any input image to be reproduced has to be adapted to the printing gamut, and the relevant processing parameters have to be selected to obtain high-quality image reproduction. This process is called the color reproduction workflow.

Color reproduction workflows known in traditional ink-based printing often rely on assumptions which are directly related to the printing of inks on paper: the printer is modeled as having a limited number of inks (most often cyan, magenta, yellow and black (CMYK) inks), each one deposited under the form of binary micro-patterns overlapping with each other. One obtains on the paper a mosaic of “colorants”, or “Neugebauer primaries” (e.g., red, green, blue), and the spectral reflectance of the print can be predicted from the individual reflectances of these colorants. The situation is very different with most laser processing techniques. As an example, in the CLM color laser marking technology, although only three different pigment layers are deposited, four colorants can be identified after laser processing [9].

In our laser processing technology, as briefly mentioned in the previous chapter, processing the same area twice results in unpredictable color changes, which cannot be accounted for by the mixing of the two colors associated to the laser parameters used for both passes. On the other hand, the number of colorants used in one image is unlimited in our case. With ink-based techniques, the number of inks is fixed, and these inks thus generate a fixed number of colorants ( $2^k$  colorants with  $k$  inks). Changing one of the inks is a cumbersome process, in which the printer has to be stopped, and which often requires a new calibration to get the printer ready for printing again. With our laser processing, every metasurface in our database can be used in the same print, as this only

requires modifying the laser processing parameters, which can be done rapidly and automatically by the laser processing software.

To account for these specificities, this chapter introduces a new color reproduction workflow allowing the reproduction of any color image with our silver-titania metasurfaces. A demonstrator showing a color image in mode “backside reflection” is produced, showing the good color reproduction capabilities of our process. However, several factors limit the quality of the print such as the reproducibility of the colors between the database and the actual image, and the border effects of the pixels. These will be a concern for both color images and multiplexed images. Strategies to overcome these limitations are proposed and will be used throughout the rest of the thesis.

#### 4.1 Color reproduction workflow

In traditional ink-based printing, the color reproduction workflow makes the link between the input digital image, and the printed physical image in a way that maximizes the quality of the reproduction. In other words, it is a way to determine what ink quantities are needed and where they need to be printed so that the final print accurately matches the desired image, accounting for the technological limitations of the printer. With color laser marking, the printer is replaced by a laser processing set-up. Instead of ink quantities, the parameters that can be controlled are the laser processing parameters : power, repetition rate, polarization interline distance and scan speed or exposure time. Instead of the ink dots that are deposited on paper, which can be superimposed and produce colors that are accurately described by well-known models, the laser modifies the material in a non-linear way. Therefore, color laser marking requires new color reproduction workflows.

As exposed in Chapter 2, several laser color marking techniques using plasmonic nanostructures exist, but their color reproduction workflows are almost never described. The only one exposed thoroughly, proposed by Cucerca et al.[65], relies on juxtaposed line halftoning, which has been derived in the framework of printing with custom metallic inks, and is applied to the laser marking of stainless steel and titanium with good color reproduction results. However, printed image multiplexing requires to take several images as input and makes the available colors dependent on the selected modes. For these reasons, we simplify this workflow by using vector error diffusion, which can perform both the halftoning and the color separation steps together and will be easier to generalize to multiplexing. Another difference is that our vectorization step generates vectorized pixels instead of lines, such as described in Section 3.1 – *Laser processing*.

Our color reproduction workflow is described in Figure 4.1. First, the input colors are retrieved from the input image, in a standard color space such as standard red green blue (sRGB). A gamut mapping step transforms the input colors to attainable colors that are inside the gamut of our laser process for the considered mode. To retrieve the

metasurfaces allowing to reproduce these colors, while preserving continuous color gradients in the image, vector error diffusion is applied. In this process, one metasurface is assigned to each of the pixels of the image. The set of laser parameters corresponding to each metasurface is then retrieved in the database created at the previous chapter, and the corresponding parameters are used to produce laser instructions that will be read by the laser processing software. During this step, which we call *vectorization*, the chosen structuring elements (dots or squares) are placed at the appropriate locations in the laser processing file.

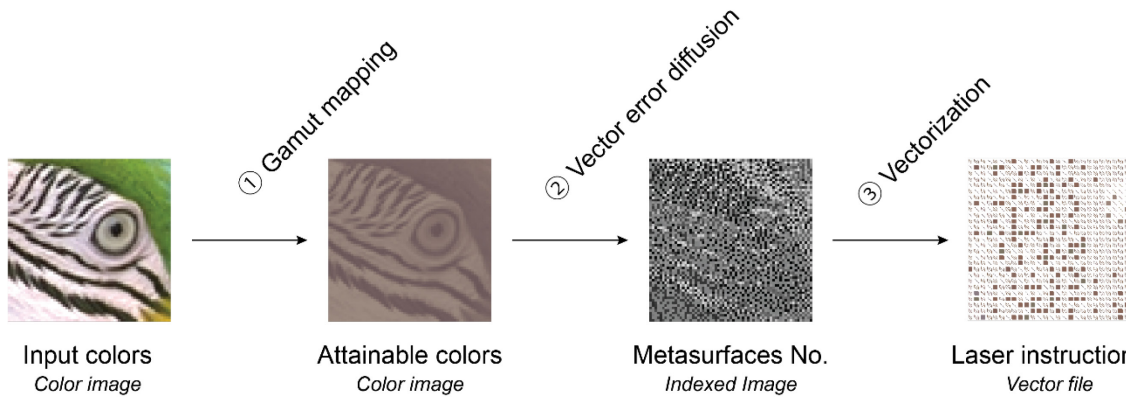


Figure 4.1 Color reproduction workflow with our laser-induced metasurfaces.

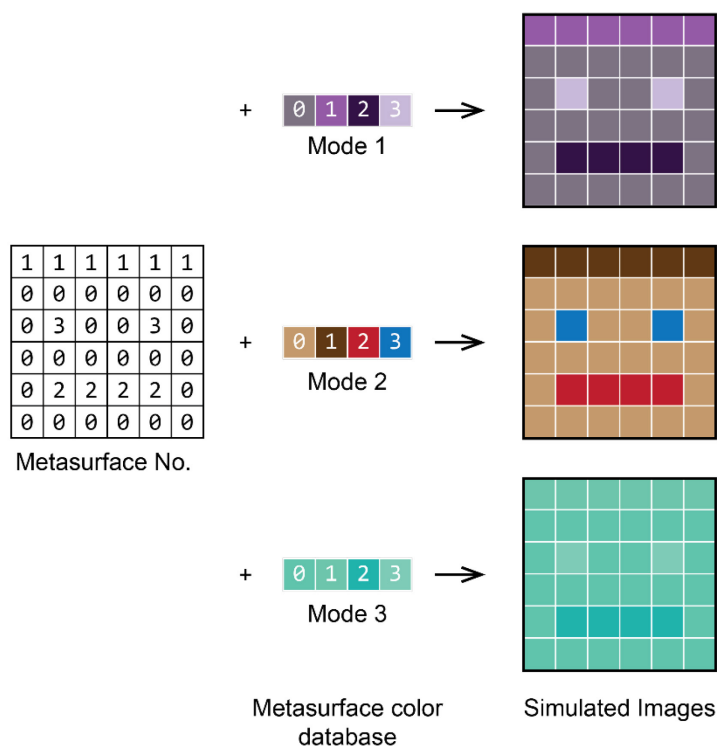
In the following, we will describe several aspects of this workflow, namely the image formats use at each step of the workflow, the gamut mapping process we are using, our halftoning process, vector error diffusion, and its ability to automatically select laser processing parameters from the database.

#### 4.1.1 Image formats

A digital image is usually encoded as an array of values describing square colored elements, called *pixels*. Each pixel is attributed either a scalar value, in the case of a grayscale image, or a vector of 3 values in the case of a color image. In the case of grayscale images, the value of each pixel encodes the lightness that will be displayed in this point. It can be encoded as a float value, varying continuously between 0 (minimum lightness) and 1 (maximum lightness) or as an integer value, in which case the number of bits used to code the value, also known as the bit depth of the image, will determine the number of lightness levels (aka gray levels) used to encode the image. The most common bit depth is 8, with integer lightness values thus ranging between 0 and 255 ( $2^8-1$ ). The smallest bit depth is 1 bit, in which case the lightness can take only two values, 0 or 1. Such an image is called a *binary image*.

For color images, when colors are encoded in a red green blue (RGB) color space such as standard red green blue (sRGB, see Appendix C, Section C.3 for a definition), the bit depth of the image can be defined as either the number of bits used to encode each value

of the 3-dimensional vectors, or the sum of these three numbers. Additionally, a fourth channel can be used to encode transparency or *alpha*, in which case each pixel will be encoded by a four-dimensional vector. The most common bit depth for color images is 8 bits per value or 32 bits total (including the alpha channel). This format will be used for the input colors in our workflow, as it accounts for most available color images. The potential alpha channel of the input image is only used for display or graphics rendering and is thus discarded here.



*Figure 4.2 Metasurface image. A single indexed image contains the number of the metasurface present in each pixel. In each mode, a different look-up table mapping each metasurface to its color in this mode is constructed from the metasurface color database. The image in each mode is simulated by displaying the indexed image with the corresponding look-up table.*

Color images can also be encoded with one scalar value per pixel (as for grayscale images) with the addition of a *colormap* which will be used as a look-up table to map the scalar values to coordinates in the chosen color space. Such an image is called an *indexed image*. In our case, this is well suited to encoding the metasurface image: the index of each pixel is the metasurface number. Then, to simulate the color image in each mode, the colormap can be set to the one extracted from the color database. When the image is displayed, the color of each metasurface will be read in the colormap and correspond to the color of this metasurface in the considered mode.



Finally, the result of our color reproduction workflow, before laser processing, is saved in a vector format. In this format, rather than defining a rectangular grid with fixed position for each pixel, each vectorized object is stored with its position. As explained previously, we base our laser processing on two structuring elements: squares and dots. The vector image format gives the liberty to define one object for each of the pixels of the image, control the laser processing parameters accordingly and control the processing order. Contrary to image formats, there are virtually as many vector formats as there are vector drawing programs, including those dedicated to laser processing. A custom Python code is used to translate the metasurface numbers into vectorized object with the correct format for our laser processing software.

### 4.1.2 Gamut mapping

Gamut mapping is the modification of the input colors to make them fit in the color gamut associated with a given output device. Gamut mapping algorithms are usually divided into two different kinds. On the one hand, gamut extension techniques are designed to expand the color range, for example in order to take advantage of a high dynamic range display when displaying a standard (low) dynamic range image. On the other hand, gamut compression techniques restrict the colors to the limited capacities of an output device such as a printer. Because our color laser marking process has a smaller color gamut than most displays, the gamut mapping algorithm used here will be a gamut compression algorithm.

The steps of most gamut compression algorithms are the following ones: first, the luminance of the input colors is adapted to the luminance range of the output gamut ; second, the chroma is modified in such manner that each input color is mapped onto a point inside the output gamut, by keeping the same hue. Depending on the algorithm, the modifications can be applied to all input colors or only to the colors outside the output gamut, or to a fraction of the colors that are close to or out of the border of the output gamut. For the luminance, different types of scaling have been studied, from a linear scaling to the output luminance range to more complex non-linear functions. The type of modification applied to the input colors can also vary from one algorithm to the other : they can be projected onto the closest point of the output gamut surface (aka. *gamut clipping*), or they can be moved closer to a projection point inside the output gamut, or even moved towards a point constructed outside the output gamut. For an exhaustive review, we refer the reader to the book by Jan Morovic[80].

When performing gamut mapping with our metasurfaces, the output gamut can be very small in certain modes. Compared to the gamuts considered in the usual gamut mapping literature, ours have several features that make using common gamut mapping algorithms impossible. First, in contrast with printing on paper where the paper white is often present in the printed image, the white metasurface has to be identified in the palette, and can sometimes be rather chromatic. A similar issue is met with the black

color, which in traditional printing is either made by a combination of all inks or a dedicated black ink, and which has to be identified in the palette in our case. Chosson and Hersch[81] described a gamut reduction technique for printing with custom inks on paper that is compatible with these limitations. The key insight we take from their work is that when no black is available in the destination gamut, the mapping should be done towards a new gray axis which connects our selected black and our selected white (instead of the standard gray axis defined in the CIE 1976  $L^*a^*b^*$  color space as  $(a^*, b^*) = 0$ ).

Our implementation of this gamut reduction algorithm is however greatly simplified in comparison to theirs, to allow its application to multiplexed images. As briefly exposed in Chapter 2, multiplexing can impose strong limitations on the available range of colors. Therefore, we designed our gamut mapping algorithm to allow its use on any selection of metasurfaces, in any mode. This selection of metasurfaces defines color gamuts associated with the different modes, to which the input colors will have to be mapped. As will be discussed in the next section, a continuous gamut can be obtained from a limited number of colors thanks to halftoning techniques, thus providing in each mode a continuous gamut characterized by the convex hull of the colors of the selected metasurfaces.

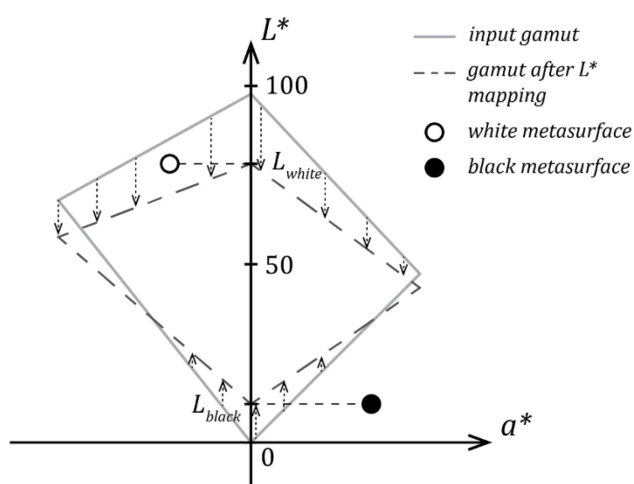


Figure 4.3 Luminance mapping of the input colors to the luminance range of the metasurface palette. Projection on the  $L^* a^*$  plane

We first perform a linear mapping of the luminance and of the chroma towards the grey axis of the palette, obtained by a translation of the colors in their respective luminance plane in the CIE 1976  $L^*a^*b^*$  color space. From the selected color palette, a white and a black are identified (respectively minimum and maximum luminance). The first step of our gamut mapping consists in a linear scaling of the input color luminance  $L_{in}$  to the luminance range  $[L_{black}, L_{white}]$  of the metasurface palette (Figure 4.3).

Maintaining the luminance contrast is crucial to represent the high spatial frequencies of an image[82]. This yields the output luminance  $L_{out}$  of the color in the mapped image:

$$L_{out} = L_{black} + \frac{L_{in}}{100}(L_{white} - L_{black}) \quad (4.1)$$

To prepare the chroma mapping step, we now transform the palette colors in such manner that the gray axis (black-white) is aligned with the gray axis of the CIE 1976  $L^*a^*b^*$  color space. This will allow to operate the chroma mapping easily in the modified space. We compute  $a'$  and  $b'$ , the modified  $L^*a^*b^*$  coordinates for each palette color, by performing a translation perpendicularly to the  $L^*$  axis such that both the white and the black are mapped to  $(a', b') = 0$ . The translation is proportional to their luminance with respect to the black-white range.  $a'$  and  $b'$  are given by the following equations:

$$a' = a - L_n^{1/p_a} * (a_{white} - a_{black}) - a_{black} \quad (4.2)$$

$$b' = b - L_n^{1/p_b} * (b_{white} - b_{black}) - b_{black} \quad (4.3)$$

where

$$L_n = \frac{L - L_{black}}{L_{white} - L_{black}} \quad (4.4)$$

$L_n$  is the normalized luminance of the considered color, and  $p_a$  and  $p_b$  are two parameters fitted in such manner that intermediate colors on the black-white axis are also mapped to the gray axis (Figure 4.4).

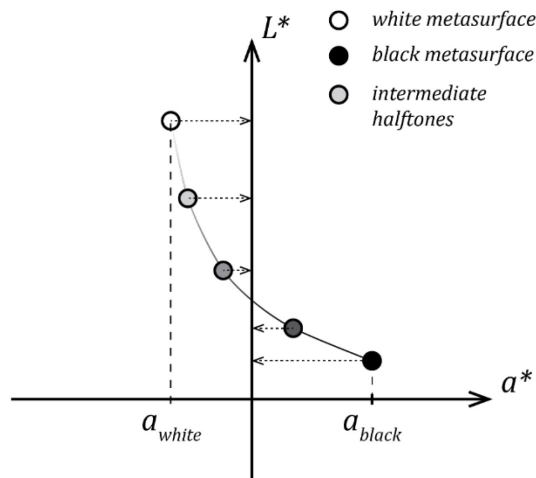


Figure 4.4 Mapping of the gray axis of the palette to the gray axis of the CIE 1976  $L^*a^*b^*$  color space

To fit them, halftones of the white and the black metasurfaces are simulated by assuming a purely additive mixing. For the mixing of two metasurfaces with respective tristimulus values  $X_1, Y_1, Z_1$  and  $X_2, Y_2, Z_2$ , the tristimulus values of the corresponding halftone are:

$$\begin{cases} X_{halftone} = xX_1 + (1-x)X_2 \\ Y_{halftone} = xY_1 + (1-x)Y_2 \\ Z_{halftone} = xZ_1 + (1-x)Z_2 \end{cases} \quad (4.5)$$

where  $x$  denotes the fractional area covered by metasurface 1 and is thus between 0 and 1. This describes a straight line in the XYZ space, but this line is curved in the CIE 1976  $L^*a^*b^*$  color space. Therefore, we fit a power function with parameters  $p_a$  and  $p_b$  so that for each halftone of white and black, we have:

$$a'_{halftone} = a_{halftone} - L_n^{1/p_a} * (a_{white} - a_{black}) - a_{black} = 0 \quad (4.6)$$

$$b'_{halftone} = b_{halftone} - L_n^{1/p_b} * (b_{white} - b_{black}) - b_{black} = 0 \quad (4.7)$$

We then divide the Luminance axis in 10 ranges  $[L_{black}, L_1], [L_1, L_2], \dots, [L_9, L_{white}]$  (Figure 4.5a) and the hue axis in 16 ranges  $[h_0, h_1], \dots, [h_{15}, h_{16}]$ . For each range, we compute the maximum attainable chroma of the metasurfaces palette in the gray-axis-aligned space  $C'_{max}$  and the maximum chroma of the input colors in this range  $C_{max}$ .

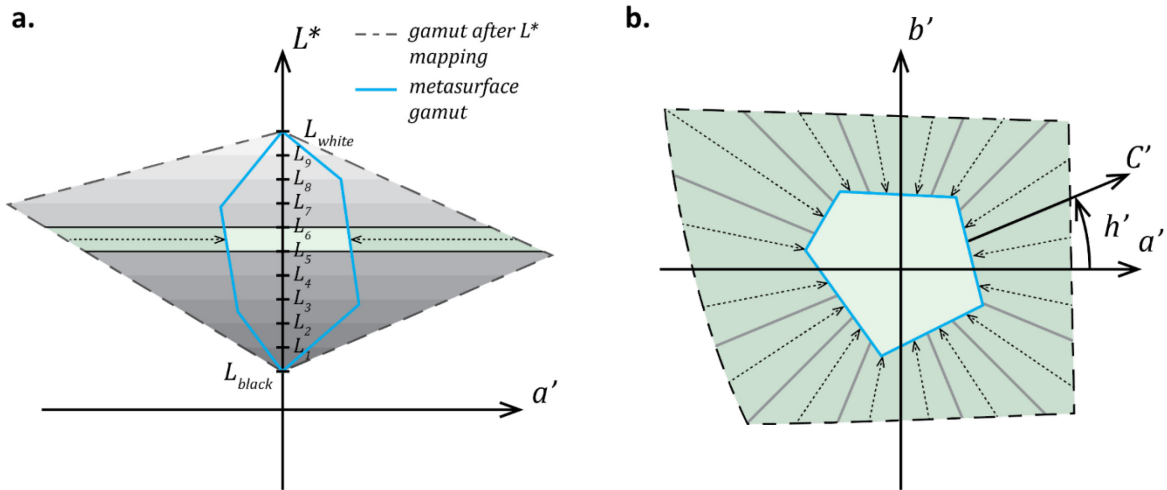


Figure 4.5 Mapping of the Chroma in the modified space. (a) The gamut is divided in 10 Luminance ranges in which the chroma mapping will be performed (b) The  $(a', b')$  plane is divided in 16 sections. In each section, the colors of the gamut are linearly translated into the metasurface gamut

We reduce the Chroma of the input colors,  $C_{in}$ , in the gray-axis-aligned space as follows (Figure 4.5b):

$$C'_{out} = \frac{C_{in}}{C_{max}} * C'_{max} \quad (4.8)$$

Finally, the colors are shifted back to align them with the original gray axis of the metasurface palette by performing the inverse transformation:

$$a_{out} = a'_{out} + L_n^{1/p_a} * (a_{white} - a_{black}) + a_{black} \quad (4.9)$$

$$b_{out} = b'_{out} + L_n^{1/p_b} * (b_{white} - b_{black}) + b_{black} \quad (4.10)$$

We obtain output colors that are included in the gamut of the metasurface palette and are thus achievable by halftoning. The result of the gamut mapping to a color palette consisting of 11 metasurfaces measured in mode “backside reflection” is shown in Figure 4.6.

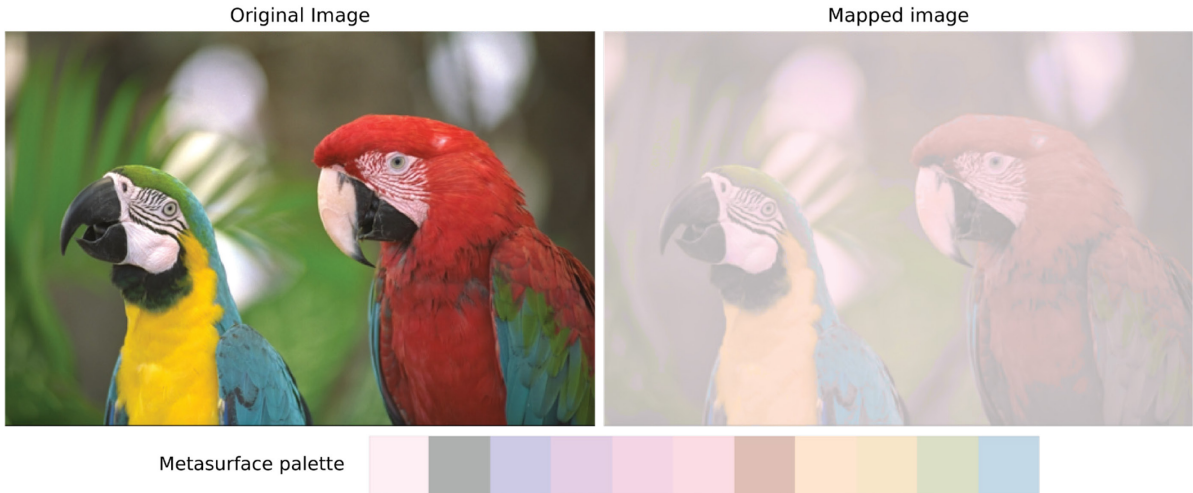


Figure 4.6 Original image and result of its gamut mapping to a palette of 11 metasurfaces in mode “backside reflection”.

### 4.1.3 Vector error diffusion halftoning

Halftoning renders continuous color gradients in an image by relying on the spatial averaging properties of human vision. It is a way of quantizing a continuous image so that it can be interpreted by a device having a discrete output. Since color halftoning algorithms were mainly created for ink-based printing, most of them rely on a superposition of dots of the available inks. In our case, however, laser-processing the same area twice modifies the color in an unpredictable way. We are therefore interested in a specific class of halftoning algorithms, which relies on a juxtaposition of the primaries rather than a superposition. Error diffusion algorithms are a kind of juxtaposed halftoning algorithm in which the error made by the quantization step is diffused to

neighboring pixels which allows to correct it thanks to the spatial averaging done by the visual system when the print is observed from a distance.

Among the error diffusion algorithms, Floyd-Steinberg dithering is the most famous and the simplest one to implement. Furthermore, it works well with low pixel count images and the number of primaries which can be used in the quantization step is not limited. It was formulated originally for binary image quantization, then extended to color according to two strategies. In the first one, the image is decomposed in grayscale channels, which are halftoned independently. However, this method is known for producing bad color reproduction[83] and is only possible when each channel can be addressed independently, such as in displays. We will not consider it further in the present work. In the second one, the process is performed simultaneously on the three channels, by considering each pixel as a vector containing its values for each channel. This is called *vector error diffusion (VED)*[84]. It has shown better color reproduction quality than the channel-by-channel versions and allows picking the primaries from a limited color palette.

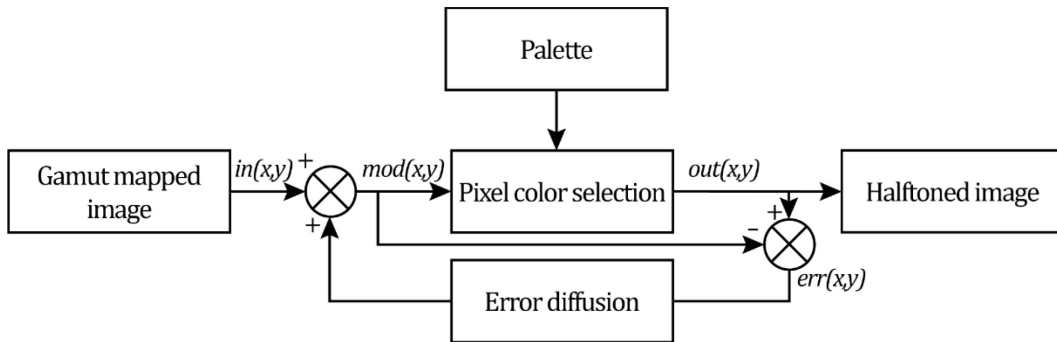


Figure 4.7 Error diffusion halftoning diagram.

The steps of the algorithm are the same for all three variations (Figure 4.7). The pixels are processed in-place, following a line-by-line raster scan. The first pixel  $in(0,0)$  is compared to the available colors in the palette and replaced by the closest one,  $out(0,0)$ . This generates a quantization error  $err(0,0)$  which is diffused to neighboring pixels according to the filter pattern defined in the algorithm (Figure 4.8). For example, the next pixel  $in(1,0)$  is modified by this error diffusion in the following way, resulting in a new pixel value  $mod(1,0)$  which will be used for the color selection step.

$$mod(1,0) = in(0,0) + \frac{7}{16}err(0,0) \quad (4.11)$$

When processing pixel  $(x, y)$ , the modified pixel value after error diffusion  $mod(x, y)$  is used for the color selection step and error computation.

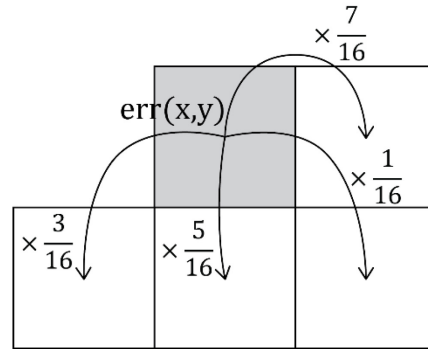


Figure 4.8. Floyd-Steinberg filter for error diffusion.

When applied to the binarization of a grayscale image, the color selection step is a simple thresholding. For example, if the grayscale image is coded with 8-bit integers, the color selection is done as follows:

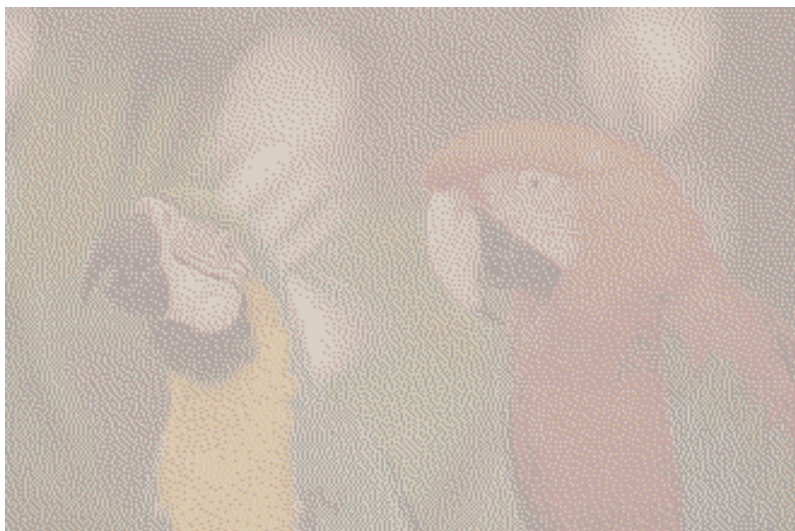
$$out(x,y) = \begin{cases} 0 & \text{if } mod(x,y) \leq 127 \\ 1 & \text{else} \end{cases} \quad (4.12)$$

In the vectorized version, all quantities are represented by vectors. The color selection step has to be performed in the vector space. The distance between the color  $mod(x,y)$  and all colors of the palette is computed, and the color having the smallest distance is selected. Then, the error is computed and diffused by element-wise operations.

Artifacts can happen when the error is big, in edges, or when colors are outside the gamut[83]. This is reduced by clipping the modified pixel value so that no negative or out of range colors are considered. In our case, this is prevented by the gamut mapping step, as all colors obtained after this step are inside the gamut.

When the error is measured in a perceptual color space that is not the device color space, a color chosen to have a small perceptual error can actually introduce a large error in the device color space, which will create artifacts[85]. For this reason, we perform all the steps of our error diffusion in sRGB. The distance chosen for the color selection step is a simple Euclidian distance.

The result of our vectorized error diffusion algorithm is that, for each pixel of the gamut mapped image, one metasurface is selected which accurately matches the local color in the gamut mapped image. This preserves smooth color gradients and produces a good quality color reproduction in the chosen mode, provided that the printing is done at a high enough spatial resolution or that the result is observed from a distance. It will also be essential to provide color gradients when only few colors per mode will be available in some multiplexed images. Figure 4.9 shows the result of the vector error diffusion step.



*Figure 4.9 Result of vector error diffusion on a previously gamut-mapped image (see Figure 4.2).*

#### **4.1.4 Parameter separation**

In traditional printing, the step of color separation transforms the image from the color space where gamut mapping was performed to the inks space (e.g. CMYK), enabling to separate the color image into grayscale images that will be halftoned and thus control the ink quantities deposited. In laser marking, colors are created not by varying ink quantities but by selecting the laser command parameters, such as power or scan speed. The color separation is therefore a matter of selecting the correct parameters to reproduce each of the desired colors, but there is not a direct link between each parameter and the color variation. Thankfully, this step is straightforward once the vector error diffusion has been performed: the parameters for each metasurface can be retrieved via a simple look-up in the metasurface database, as metasurfaces have already been selected in the pixel color selection step.

## **4.2 Experimental results and limitations**

To demonstrate the validity of our color reproduction setup, we inscribed the image shown in Figure 4.9 to be observed in mode “backside reflection”. Figure 4.10 shows the resulting printed image as observed with our color measurement setup. The color reproduction is fairly good, and visual observation of the sample by several observers confirms this. This result confirms the interest of this technology for color image marking. Furthermore, several experimental limitations can be highlighted from it, which will have to be taken into account in all further experimental demonstrations of this work. These will be discussed in the following, as well as the solutions implemented to overcome them.





Figure 4.10 Demonstrator of a single color image printed with our laser color marking technology. The size of the printed image is 23.8 mm x 15.84 mm. Each printed pixel is a  $72 \times 72 \mu\text{m}^2$  square. Image taken with our imaging setup in mode “backside reflection”.

Table 4.1. RGB values (8 bits) measured on the 12 metasurfaces shown in Figure 3.7 and standard deviation calculated from the measurements carried out on 7 samples printed with the same laser parameters in modes Backside Reflection and Unpolarized Transmission (in brackets). The low standard deviation shows the good reproducibility of the laser processing.

Square Name	Backside Reflection			Unpolarized Transmission		
	R (std)	G (std)	B (std)	R (std)	G (std)	B (std)
A1	171 (4)	169 (4)	147 (4)	223 (1)	213 (1)	205 (1)
A2	207 (3)	172 (3)	135 (4)	202 (1)	170 (1)	163 (1)
A3	203 (9)	187 (9)	155 (11)	217 (4)	198 (3)	182 (2)
A4	221 (4)	155 (3)	147 (1)	191 (2)	162 (1)	152 (0)
A5	208 (3)	167 (2)	141 (2)	218 (1)	196 (1)	172 (2)
A6	199 (2)	167 (2)	179 (1)	182 (1)	151 (1)	133 (1)
B1	154 (2)	156 (4)	155 (6)	218 (1)	200 (1)	185 (1)
B2	200 (4)	183 (4)	139 (3)	161 (2)	138 (2)	166 (2)
B3	179 (2)	177 (2)	155 (1)	213 (3)	193 (3)	178 (2)
B4	228 (2)	189 (1)	152 (1)	163 (3)	125 (2)	136 (2)
B5	169 (2)	148 (2)	127 (2)	202 (1)	174 (1)	155 (1)
B6	206 (1)	162 (1)	170 (1)	179 (1)	154 (1)	135 (1)

### 1.1.1 Reproducibility of the colors

As several steps of the preparation process are still manual, reproducibility between different samples is not perfect. As a result, a color variation between metasurfaces inscribed with the same laser parameters on two different samples can be observed. Therefore, when the color reproduction workflow is performed using the colors of the database, a slight color shift can be observed in the resulting print.

However, reproducibility within the same sample is good. There is a good accordance between the colors of two metasurfaces inscribed with the same laser parameters on the same sample. Table 4.1 relates the color coordinates measured for modes “transmission” and “reflection” on the metasurfaces of Figure 3.7 and their standard deviation measured on 6 reproductions of the same metasurfaces, demonstrating good color repeatability within one sample.

To ensure the best results, all or part of the database is thus reproduced and measured on each new sample, and the steps of the image reproduction workflow described above are done with the updated colors before the laser inscription of each new image.

### 4.2.1 Interference patterns

Adding to the potential variations of color from one sample to another, certain regions of the writing field display fringes pattern which might be due to interferences between the incident laser beam and its reflection on the backside of the glass slide. This effect, probably due to a variation in the local deposited energy, can be seen on some of the metasurfaces which color is sensitive to small variations of power. Figure 4.11 highlights such an effect on the printed color image. When possible, these regions of the writing field will be avoided for color measurements. However, this local effect does not hurt the overall perception of the printed images.

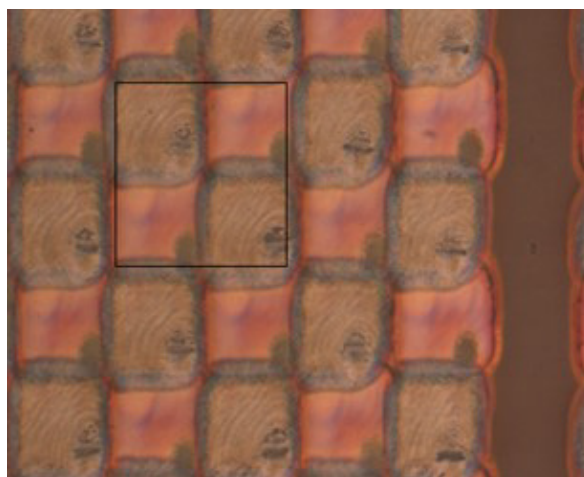


*Figure 4.11. Detail of the printed color image where fringes are perceptible, circled with dashed white lines.*

### 4.2.2 Pixel overlap

In our color reproduction workflow, the pixels are assumed to be printed side by side so that the mixing between two pixels is only due to an averaging due to the visual system of the observer. However, when the laser-affected area is large for one given set of parameters, the borders of one pixel can overlap with those of the neighboring pixel. This happens when the power or the number of pulses per unit area is high.

Figure 4.12 shows such an example where two square pixels are printed together in a 50% halftone. The expected size of these pixels is  $40^2 \mu\text{m}^2$ , however, due to a high power and number of pulses, the affected area is bigger. This results in an overlapping between the laser processing of the two pixels. In this particular case, we can see that the overlapping produces a blueish color, closer to one of the two pixels. If we take a look at a  $2 \times 2$  pixel tile ( $80^2 \mu\text{m}^2$  square), the areas colored in pink (color of the smallest pixel) and in blue (color of the biggest pixel) should be equal. However, it is clear that the blue area is bigger than the pink area. This will result in the halftone having a different color than expected, probably more blueish.



*Figure 4.12. Pixel overlap example: microscopic image of two metasurfaces printed in a 50% halftone with  $40 \times 40 \mu\text{m}$  pixel size. The black square shows an  $80 \times 80 \mu\text{m}$  area in which the respective area coverages of both pixels should be equal.*

As mentioned previously, there is no model capable of predicting the color of the overlapping areas. Thus, to prevent issues due to overlapping, only low powers or number of pulses are used. However, knowing for sure that two pixels will be correctly juxtaposed is impossible. Indeed, although the colored area around a single pixel can indicate the laser affected area, nothing indicates for sure that it will affect neighboring pixels. To check for this problem, a halftoning matrix is inscribed by printing 50% halftones of every couple of metasurfaces in the palette, and the resulting print is measured with the color characterization setup. If the color of the halftone is close to the one expected (average of the two metasurface colors), then the metasurfaces can be used

to produce an image. When the number of metasurfaces is not too high, this provides an effective way of checking that there is no overlapping issue.

Another way to overcome this is to use bigger pixels. This is only possible with square pixels, as dot pixels have a size that only depends on the laser-affected area. Although this reduces the final resolution of the print, it makes the overlapping area smaller compared to the center area of the pixels, which allows for a much bigger overlapping without changing the average color dramatically. For this reason, most of the following demonstrators were realized with  $72 \times 72 \mu\text{m}$  pixels to ease the metasurface selection process.

### 4.2.3 Limited pixel count

The use of vectorized format for the control of the laser is beneficial for its ease of use and its versatility. However, vector files have a size that increases with the number of objects. This has a direct impact on the performance of laser processing software associated to our nanosecond laser, to a point where opening one file can take up to minutes. This limits the number of objects that can be used practically, which in our cases limits the pixel count of our images. Therefore, a tradeoff has to be found between the size and resolution of the image to keep the pixel count around a maximum of 100 000. Although this can probably be overcome in further industrial implementations with a dedicated laser processing software, this limits the quality of the demonstrators produced in this thesis.

## 4.3 Conclusion

The colors of our Ag:TiO<sub>2</sub> metasurfaces produced by nanosecond laser irradiation can be used to produce any arbitrary color image using an adapted color reproduction workflow. Although perfect color reproduction is not targeted, the gamut mapping strategy exposed here preserves the color of the original image while allowing to use very limited palettes where the usual achromatic black and white are not present. This is done by identifying the metasurface colors being closest to black and white in the palette and mapping all other colors in respect to this modified gray axis. This can be performed on any reduced color palette with at least two colors. Then, vectorized error diffusion allows performing both halftoning and parameter separation in an efficient and flexible way. Due to the small size of our samples (a few square centimeters) and the limited pixel count imposed by the laser processing software, its ability of producing high-quality halftoning even for low-resolution images is highly beneficial.

We demonstrate the printing of an arbitrary color image with this workflow using our technology. Even though it was not the main objective of this work, this opens the way for the laser marking of a single color image with Ag:TiO<sub>2</sub> metasurfaces with a

nanosecond laser. It also highlights some of the limitations of our laser processing technology encountered when producing images:

- First, the color reproducibility between samples is not good enough to use a metasurface color database measured on one sample for color reproduction on another sample. Once the color palette has been selected from the metasurface database, it has to be inscribed and measured again on the current sample to take any possible change in color into account. In the color reproduction workflow, only the colors obtained on the current sample are used for the metasurface selection step, in order to ensure the best result.
- Second, the size of the inscribed pixel must be kept close to the chosen pixel size in order to make sure the halftoning produces the right colors. Otherwise, overlapping appears which creates unpredictable colors. For this, when possible, the power and number of pulses are kept low, so that the laser-affected area is small, which in turn keeps the size of the inscribed pixels to a minimum. However, reducing the power and number of pulses is not always possible as it reduces the range of available metasurfaces and thus the color gamut. To be able to include higher power and number of pulses, the size of the pixel can be increased, which reduces the relative importance of overlapping effects.
- The last limitation is due to the vector format used by the laser software coming with our laser, which limits the pixel count in the images that can be printed due to performance issues.

These limitations are technology-related and would be solved in an industrial implementation. For the purpose of this thesis work, some workarounds have been put in place to overcome them. As they apply to the production of any image, these workarounds will be helpful in the realization of multiplexed images as well, as is the purpose of the next three chapters.

In these, three different algorithms are described to find the sets of metasurfaces that can be used for printed image multiplexing. Once these sets of metasurfaces are identified, the steps discussed in this chapter, i.e., gamut mapping, vector error diffusion, and vectorization/parameter separation, have to be applied as well to produce visually appealing images. The variations of this color workflow to accommodate each kind of multiplexing will be described in the corresponding chapters. In Chapter 5 and Chapter 6, the selection of the color palettes allows to perform this workflow on each image independently, while Chapter 7 presents a generalization of this workflow to be directly applicable to all multiplexed images at the same time.



## Chapter 5.

# Conditional gamut algorithm

The previous chapter described the printing of a single image using our metasurfaces. Based on our metasurface colors database, or on a subset of these metasurfaces colors, we are able to adapt any image in order to be reproducible with our metasurfaces. By implementing vector error diffusion, we are able to select for each pixel the metasurface whose color is closest to the color expected in the image.

Our objective is now to produce multiplexed prints displaying different images in two or more modes. The choice of the metasurfaces should not only be based on their color in one mode, but on their respective colors in all the considered modes. The set of images that we want to display in these modes determines, in each pixel, the set of colors that the metasurface is expected to display, and the question is to know whether this metasurface exists in the database. Since we are constrained by the optical properties of the metasurfaces that we are able to produce, we are far from certain that all the color sets needed to reproduce the wanted images can be achieved. Changing the images in order to be sure that they are displayable with realizable metasurfaces is precisely the issue that we address in this chapter.

We will first consider the multiplexing of binary images, which are arguably the most basic form of image: only two colors are used to convey information. Accordingly, multiplexing can be achieved if we are able to display in each mode the two colors, independently on the colors wanted locally in the other modes. The required conditions form a combinatorial problem, whose complexity increases dramatically with the number of metasurfaces. We therefore propose to reduce this complexity by clustering the metasurfaces into sets of metasurfaces that have similar colors. A custom search algorithm then allows to identify the metasurfaces able to display the wanted colors in the different modes. Although multiplexing binary images is the main objective of this chapter, the possible extension of this algorithm to more colors is also discussed. We will show in particular a demonstrator displaying three binary images using Ag:TiO<sub>2</sub> metasurfaces created by nanosecond laser processing, the first three-image multiplexed print ever exhibited for which all images are visible with the naked eye and no other tool. We also demonstrate the additional possibilities of adding multiplexed details in parts of an image, or multiplexing a fourth blank image conceals the others in one of the modes

for added security. Most of these results have been published in a peer-reviewed article[78]. Here, more explanations are added to each step of the search for multiplexing solutions, as well as a discussion on how the quality of the demultiplexed images is optimized.

## 5.1 Conditions for multiplexing

Let us consider first the case of binary images, which contain only two colors. In image multiplexing, the different images are combined to form a multiplexed image and they must be revealed independently in each corresponding mode. Therefore, in each mode, the metasurfaces of the multiplexed print have to display exactly two different colors, to reconstruct each binary image. The binary images are considered to be independent, which means that the positions of the two colors in each image is uncorrelated. Therefore, two pixels that display the same color in one mode may be expected to display different colors in the other modes.

Let us first consider two modes. In the example illustrated in Figure 5.1, Figure 5.1a shows the images to be displayed, and Figure 5.1b the multiplexed print containing four metasurfaces. The images in Figure 5.1a have been designed to be completely uncorrelated. Each image is binary and should be displayed in two colors, which are located in two separate areas. The multiplexed print is constructed by intersecting these areas with those of the other image: this defines four areas on the multiplexed print that will each display a different combination of colors, one in each mode.

These combinations of colors can be represented in a logical color tree, as illustrated in Figure 5.1c. Implementing printed image multiplexing requires finding out, among the full set of metasurfaces produced by laser processing, specific sub-sets that form such a logical color tree in the selected observation modes. The logical color tree in Figure 5.1c can be read as follows: “In mode 1, metasurfaces MS1 and MS2 display the same color ■ (orange) whereas metasurfaces MS3 and MS4 display the same color ■ (blue) ; In mode 2, metasurfaces MS1 and MS4 display the same color ■ (red) whereas metasurfaces MS2 and MS3 display the same color ■ (green).”

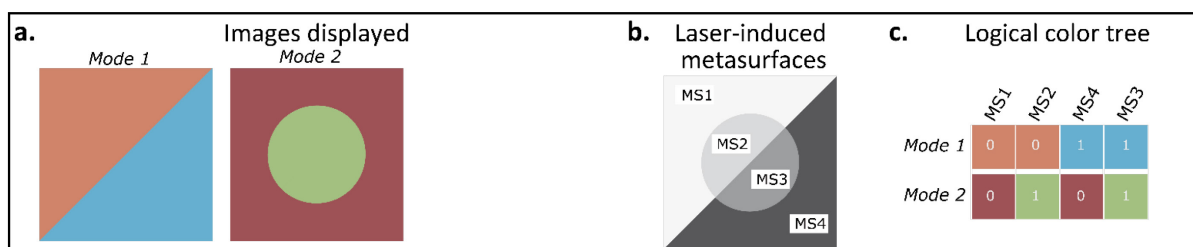


Figure 5.1 Principle of image multiplexing for 2 modes (a) 2 bicolor images to be displayed in mode 1 and 2. (b) 4 laser-induced metasurfaces (MS) are needed to multiplex the two images. (c) The colors of the 4 metasurfaces in modes 1 and 2 form a logical color tree with 2 modes and 2 colors per mode.



When three images are multiplexed, a third mode is added to the two first ones, in which two distinct colors must also be displayed. Figure 5.2a shows three images to be multiplexed in three modes. All solid color areas intersect those of the other images. Therefore, the multiplexed print is divided in eight areas with eight metasurfaces (Figure 5.2b). The corresponding logical color tree is shown in Figure 5.2c and gathers the eight color combinations needed to produce this multiplexed print. The following conditions on the metasurfaces can be read from it: “In mode 1, metasurfaces MS1, MS2, MS3 and MS4 display the same color ■ (orange) whereas metasurfaces MS5, MS6, MS7 and MS8 display the same color ■ (blue) ; In mode 2, metasurfaces MS1, MS2, MS5 and MS6 display the same color ■ (red) whereas metasurfaces MS3, MS4, MS7 and MS8 display the same color ■ (green) ; In mode 3, metasurfaces MS1, MS3, MS5 and MS7 display the same color ■ (purple) whereas metasurfaces MS2, MS4, MS6 and MS8 display the same color ■ (yellow).”

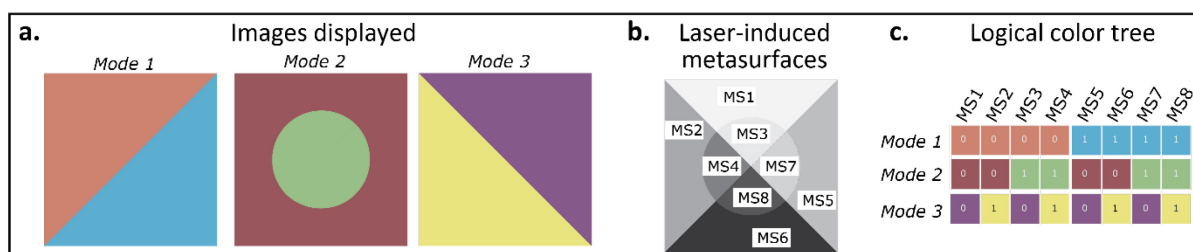


Figure 5.2 Principle of image multiplexing for 3 modes (a) 3 bicolor images to be displayed in mode 1, 2 and 3. (b) 8 distinct metasurfaces are needed to multiplex the three images. (c) The colors in modes 1, 2 and 3 of the 8 metasurfaces form a logical color tree with 3 modes and 2 colors per mode.

Expanding this logical color tree structure allows to derive the conditions for more modes and more colors per mode. When selecting  $M$  modes and  $K$  colors per mode, a logical color tree gathers  $K^M$  types of metasurfaces that provide all the possible combinations of the  $K$  colors in the  $M$  modes. The example shown in Figure 5.1 corresponds to the case where  $K = 2$  and  $M = 2$  (4 metasurfaces) and the example shown in Figure 5.2 corresponds to the case where  $K = 2$  and  $M = 3$  (8 metasurfaces).

Figure 5.1c and Figure 5.2c.

To be able to implement multiplexing, we must find in the metasurface database the metasurfaces that will encode each color combination.

## 5.2 Definitions

Even for a low number of colors and modes, testing all the sets of  $K^M$  metasurfaces among the whole database is not convenient and generates a combinatorial explosion. If  $n$  is the number of metasurfaces in the database, the listing of the combinations of  $K^M$  metasurfaces alone requires  $C_{K^M}^n = n! / (K^M! \times (n - K^M)!) = \mathcal{O}(n^{K^M})$  operations. We rather propose to identify first the colors and their combinations present in the database before finding which metasurfaces enable them, in an attempt to reduce the complexity of the search. This will be done using clustering, which we define here, and allows us to expose two concepts related to multiplexed colors: hypercolors and conditional gamuts.

### 5.2.1 Clustering

Approaching this problem in terms of discrete colors and color combinations such as what is described in a logical color tree requires a way of grouping the metasurfaces that display the same color in a given mode. The problem of identifying discrete colors from a continuous color space is known to have no good answer[86]. However, numerous color quantization techniques exist and have been applied to the color palettization problem, which aim is to find optimum colors to represent images on a color-limited device[87]–[89].

Hierarchical clustering creates clusters by performing successive pairwise comparisons. Clusters first consist of a single element and are then merged with the clusters that minimize a metric (aka distance) which defines the kind of hierarchical clustering that is performed. In *complete linkage* or *maximum distance* clustering, the distance between two clusters is the maximum distance between two elements taken in these clusters. Therefore, stopping the clustering at a defined cutoff distance ensures there is no pairs of elements within a cluster that are distant from each other by more than the cutoff value. If  $X$  and  $Y$  are two clusters,  $d$  is the distance between two elements,  $D$  the clustering metric and  $D_{cutoff}$  is the cutoff distance, the clusters  $X$  and  $Y$  are merged only if the following condition is met:

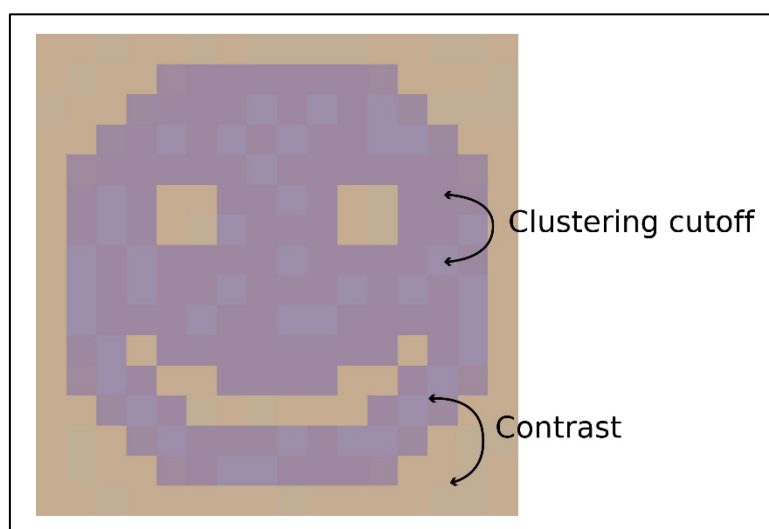
$$D(X, Y) = \max_{x \in X, y \in Y} d(x, y) < D_{cutoff} \quad (5.1)$$

This results in clusters where the distance between two elements is bounded by  $D_{cutoff}$ .

Provided that metasurface colors are represented in a uniform color space such as the CIE 1976  $L^*a^*b^*$  color space, and  $d$  is a perceptual distance in this space, performing a complete linkage clustering with a cutoff distance set to the perceptibility threshold of color differences (also called *just noticeable color distance*, *JND*) should form clusters containing only *metameric* metasurfaces, i.e., metasurfaces that display the same color in this mode. However, we have allowed ourselves some freedom in the choice of this cutoff value, since the just-noticeable difference in our case can only have an approximative

meaning, for the following reasons. As exposed in Chapter 3, color perception with the surfaces and lighting conditions that we use is not as well established as the color perception of diffusing supports. Moreover, the perception of colors in an image is much more complex than the one of large solid color areas surrounded by black or white, which are usually considered in colorimetry. Lastly, the color measurement setup that we use is not as precise as the spectrophotometers used to measure the spectral reflectance of paper prints. All of this justifies that we rather optimized empirically the cutoff value as well as the chosen distance.

Two distances were experimented in this study : CIE DE 2000, and  $\Delta L^*$  which is the difference of luminances in CIE 1976  $L^*a^*b^*$  color space (computations for both distances are exposed in Appendix C). The former creates clusters of metasurfaces that have the “same” color, while the latter creates clusters of metasurfaces that have the same lightness (but possibly different hues and chromas). The cutoff distance is varied from 1 to 10 for both distances, and the optimum was found to be dependent on the result of the algorithm. Indeed, in the final multiplexed images, if the contrast (i.e. the colorimetric distance between the pairs of colors chosen by the algorithm) is high, the observers are less sensitive to a small color difference between two metasurfaces that should have the same color.



*Figure 5.3 Clustering cutoff vs image contrast. Clustering cutoff controls the intra-cluster variability, which results in color variability within the sections of the image that should display the same color. Contrast is measured between the average color of each of the two clusters. Similar to a signal-to-noise ratio, the ratio of contrast to clustering cutoff must be kept high in order to make sure the image is correctly displayed.*

To illustrate this, Figure 5.3 simulates a binary image created by virtually selecting two clusters to reproduce a binary image. The two regions of the image that should display the same color are produced by metasurfaces from different clusters. However,

due to a large clustering cutoff, in each region, the metasurfaces can induce a slight color difference from one pixel to another. It will not disturb the viewing of the image as long as the contrast between the two separate areas (smiley face or background) is much higher than the contrast between different metasurfaces of the same cluster. Similarly, the clustering cutoff can be increased if it enables to find multiplexing solutions with a higher contrast.

Clustering also allows to represent programmatically two concepts that are crucial for understanding the implications of multiplexing: hypercolors, and conditional gamuts.

### 5.2.2 Hypercolors

Considering a set of  $M$  clusters, one per mode in  $M$  modes, gives a  $M$ -dimensional color, which we call a *hypercolor*. In the logical color trees of Figure 5.1c and Figure 5.2c, the columns actually define different hypercolors. The multiplexing is possible if each hypercolor is produced by at least one metasurface. Several metasurfaces can produce the same hypercolor, and choosing between these can allow selecting the best one in terms of laser processing (e.g. writing speed) or reproducibility. If one of the metasurfaces of a multiplexed print causes one of the printing problems described in Section 4.2 - Experimental results and limitations (big affected area overlapping with other pixels, or sensitivity to small intensity variations leading to interference fringes), one can choose from any of the other metasurfaces producing the same hypercolor to replace the problematic one in a further print.

Reasoning in terms of hypercolors can help to reduce the number of solutions that have to be tested. Indeed, several metasurfaces can produce the same hypercolor. If they were each considered independently, each one would create one multiplexing solution for each combination with the metasurfaces that produces the other hypercolors. This can rapidly end up creating millions of solutions that in fact provide exactly the same solution for multiplexing. Considering hypercolors prior to metasurfaces can therefore reduce greatly the complexity of the search.

The metasurfaces producing the same hypercolor will be found in the intersection of the  $M$  clusters constituting this hypercolor. This allows to consider only the displayed colors for the search of solutions, before picking the metasurfaces in a second step.

### 5.2.3 Conditional gamuts

By accessing the metasurfaces in a given cluster (i.e. the metasurfaces that display the same color in one mode), one can explore the colors displayed by only these metasurfaces in the other modes. For this color in the first mode, the range of colors accessible in the other modes is called the *conditional gamut*[90]. Conditional gamuts are a powerful tool to evaluate the possibilities of multiplexing, since the constraints that multiplexing imposes can be viewed as a problem of conditional gamut intersections. They are the basis of the multiplexing algorithm discussed below.

### 5.3 Conditional gamut algorithm

The following search algorithm is proposed to find all the possible solutions that enable multiplexing within a database such as the one presented in Chapter 3. It is based on the conditional gamut concept introduced above. To ease its understanding, we describe and illustrate the algorithm in the case of two and three modes. For illustration purposes, we represent the colors of the metasurfaces as points on a virtual 2D plane. In this representation, clusters will be represented as 2D areas indicated by a potato shape. This will allow the reader to keep track of the intersections mentioned in the algorithm, however, these intersections are to be understood as set intersection rather than the areas intersection only used to represent them.

#### 2 modes

Here, both the number of modes  $M$  and the number of colors per mode  $K$  are set to 2, which corresponds to the case where we want to multiplex 2 binary images.

#### *Algorithm 1 Conditional gamut algorithm for 2 colors and 2 modes*

- 
- 1 The colors of the  $N$  laser-processed metasurfaces are retrieved from the database.
  - 2 In each mode, a maximum distance hierarchical clustering is performed. Let  $C_i^m$  be the cluster label  $i$  in mode  $m$ .
  - 3 For each pair of selected modes:
    - 4 For each pair of clusters in mode 1,  $C_i^1$  and  $C_j^1$ , which contain at least two metasurfaces each, identify all the clusters  $C_k^2$  in mode 2 that contain at least one metasurface belonging to  $C_i^1$  and one metasurface belonging to  $C_j^1$ .
    - 5 If there are at least two clusters  $C_k^2$  and  $C_l^2$  that satisfy the condition of step 4, then any combination of 4 metasurfaces that belong respectively to the four sets:  $S_1 = C_i^1 \cap C_k^2$ ,  $S_2 = C_i^1 \cap C_l^2$ ,  $S_3 = C_j^1 \cap C_k^2$  and  $S_4 = C_j^1 \cap C_l^2$  is a solution for two-image multiplexing with two colors per mode.
  - 6 The solutions of step 5 are filtered based on the contrast between their respective clusters in each mode.
  - 7 For each combination of clusters identified in step 6, the best combination of metasurfaces is selected.
- 

Figure 5.4 illustrates the different intersections considered at each step of the algorithm. To keep the drawing simple, the clusters and modes which are represented are all part of a multiplexing solution. In practice, the algorithm iterates on all pairs of considered modes, and many clusters are considered at each step of the algorithm.

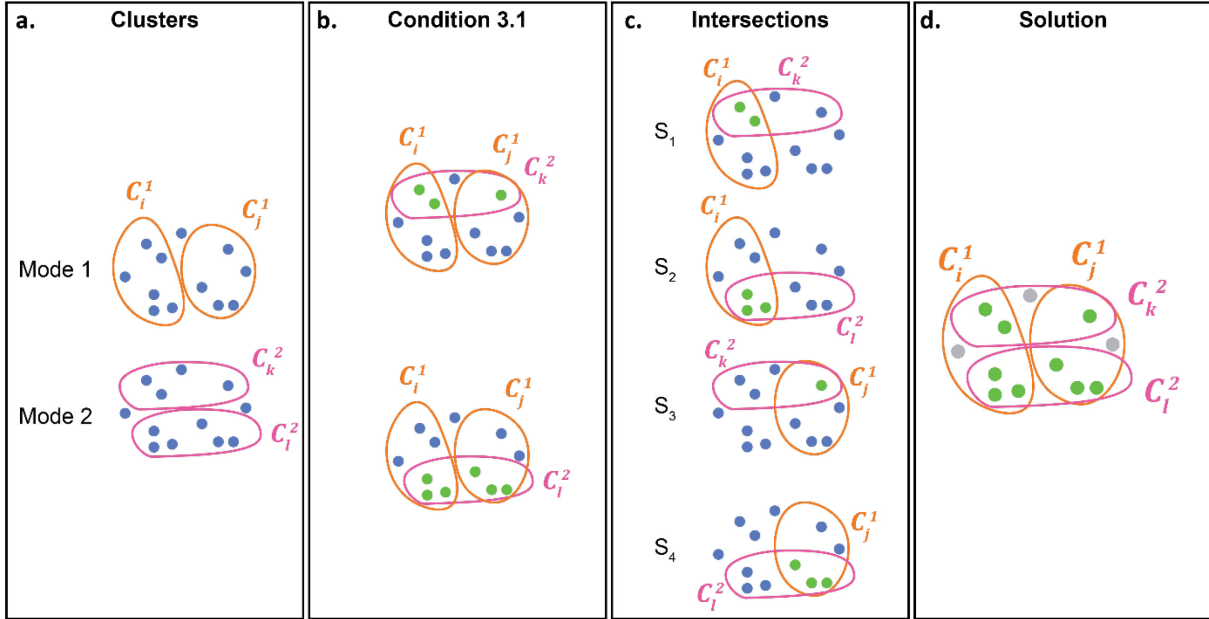


Figure 5.4 Conditional gamut algorithm for 2 modes: cluster intersections and resulting multiplexing solution. The blue dots represent metasurfaces in the database on which the algorithm is applied. (a) Four clusters obtained after step 2.  $C_i^1$  and  $C_j^1$  are selected in step 3.1 as they contain at least two metasurfaces. (b) Clusters  $C_k^2$  and  $C_l^2$  are selected in mode 2 because they comply with the condition formulated in step 3.1 (there is at least one metasurface in each of their intersections with clusters  $C_i^1$  and  $C_j^1$ ). The metasurfaces considered for the evaluation of this condition are shown in green. (c) The four intersections  $S_1 = C_i^1 \cap C_k^2$ ,  $S_2 = C_i^1 \cap C_l^2$ ,  $S_3 = C_j^1 \cap C_k^2$  and  $S_4 = C_j^1 \cap C_l^2$  as considered in step 3.1.1. Since there is at least one metasurface in each intersection  $\{C_i^1, C_j^1, C_k^2, C_l^2\}$  forms a multiplexing solution. (d) The metasurfaces colored in green can be used to produce two-image multiplexing because they belong to the intersections  $S_1, S_2, S_3$  and  $S_4$ .

When a cluster  $C_i^1$  is selected in mode 1, all the clusters of mode 2 that contain the metasurfaces of  $C_i^1$  form the conditional gamut associated to  $C_i^1$  in mode 2. The solutions for multiplexing are found when the conditional gamut associated to  $C_i^1$  and the conditional gamut associated to  $C_j^1$  have at least two clusters in common. Actually, the condition in step 4 imposes that the considered clusters of mode 2 are all in this intersection. This idea of reducing the search to the intersections of the conditional gamuts speeds up the search process and gives its name to this algorithm.

The filtering and metasurface selection of steps 5 and 6 will be detailed in the following.

### 3 modes

The conditional gamut algorithm for 3 modes is similar to the one for 2 modes with the exceptions that an additional step is required to account for the conditions of the third mode and a higher number of metasurfaces is required inside the cluster intersections at

each intermediate step. The algorithm is described in the following. Differences with the case where  $M=2$  are written in **bold**.

*Algorithm 2. Conditional gamut algorithm for 2 colors and 3 modes*

- 
- 1 The colors of the  $N$  laser-processed metasurfaces are retrieved from the database.
  - 2 In each mode, a maximum distance hierarchical clustering is performed. Let  $C_i^m$  be the cluster label  $i$  in mode  $m$ .
  - 3 For each **triplet** of selected modes:
  - 4 For each pair of clusters in mode 1,  $C_i^1$  and  $C_j^1$ , which contain at least **four** metasurfaces each, identify all the clusters  $C_k^2$  in mode 2 that contain at least **two** metasurface belonging to  $C_i^1$  and **two** metasurface belonging to  $C_j^1$ .
  - 5 If there are at least two clusters  $C_k^2$  and  $C_l^2$  that satisfy the condition of step 4 with at least **two** metasurfaces per cluster intersection, identify all the clusters  $C_m^3$  in mode 3 that contain at least one metasurface belonging to each cluster intersection  $S_1, S_2, S_3$  and  $S_4$ . two-image multiplexing with two colors per mode.
  - 6 If there are at least two clusters  $C_m^3$  and  $C_n^3$  that satisfy this condition, then any combination of 8 metasurfaces that belongs respectively to the eight sets:  $R_1 = S_1 \cap C_m^3$ ,  $R_2 = S_1 \cap C_n^3$ ,  $R_3 = S_2 \cap C_m^3$ ,  $R_4 = S_2 \cap C_n^3$ ,  $R_5 = S_3 \cap C_m^3$ ,  $R_6 = S_3 \cap C_n^3$ ,  $R_7 = S_4 \cap C_m^3$  and  $R_8 = S_4 \cap C_n^3$  is a solution for three-image multiplexing with two colors per mode.
  - 7 The solutions of step 6 are filtered based on the contrast between their respective clusters in each mode.
  - 8 For each combination of clusters identified in step 7, the best combination of metasurfaces is selected.
- 

Figure 5.5 represents the intersections considered at step 3.1.1.1. Previous steps are not represented as they are identical to the case where  $M=2$ .

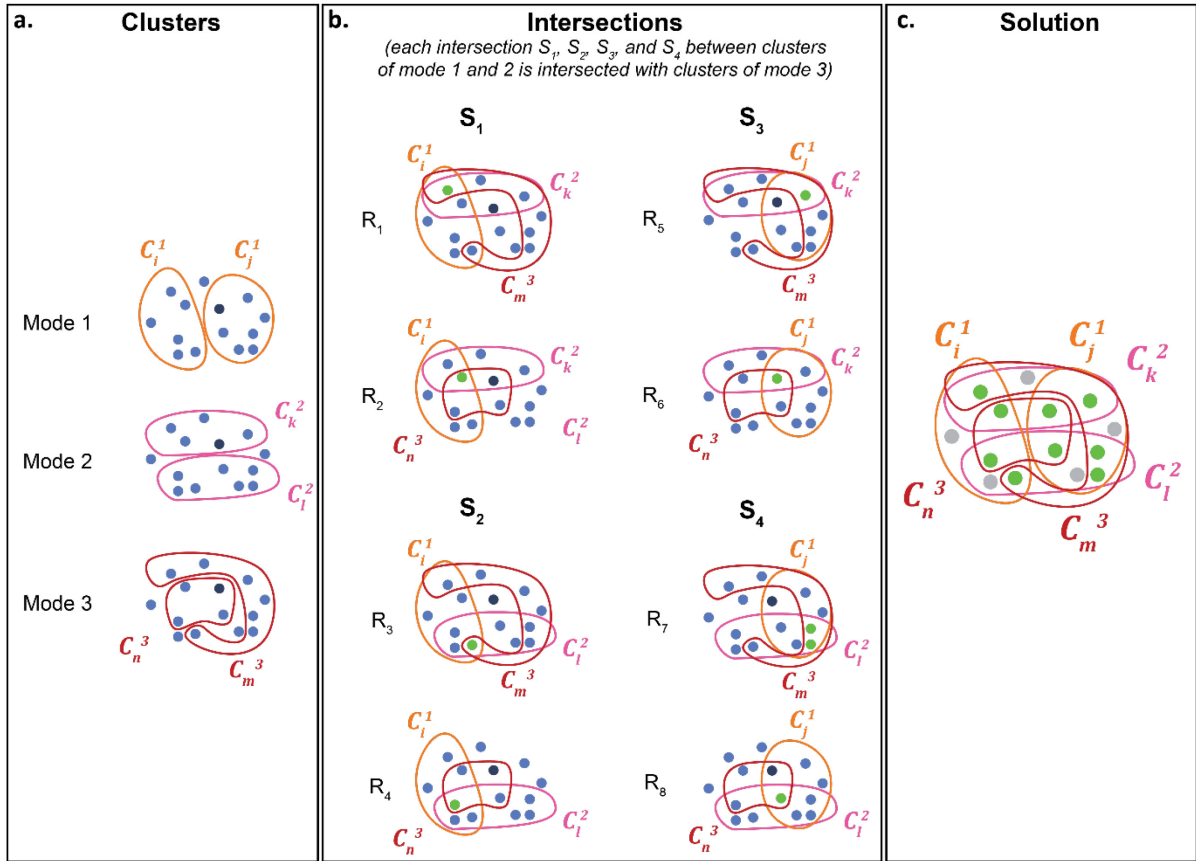


Figure 5.5 Conditional gamut algorithm in 3 modes: additional cluster intersections considered and resulting multiplexing solution. (a) Six clusters obtained after step 1.  $C_i^1$ ,  $C_j^1$ ,  $C_k^2$  and  $C_l^2$  are identical to Figure 5.4. One metasurface has been added to allow for a three-mode multiplexing solution (colored in dark blue). (b) The eight intersections  $R_1 = S_1 \cap C_m^3$ ,  $R_2 = S_1 \cap C_n^3$ ,  $R_3 = S_2 \cap C_m^3$ ,  $R_4 = S_2 \cap C_n^3$ ,  $R_5 = S_3 \cap C_m^3$ ,  $R_6 = S_3 \cap C_n^3$ ,  $R_7 = S_4 \cap C_m^3$  and  $R_8 = S_4 \cap C_n^3$  as considered in step 3.1.1.1. Since there is at least one metasurface in each intersection,  $\{C_i^1, C_j^1, C_k^2, C_l^2, C_m^3, C_n^3\}$  forms a multiplexing solution. (d) The metasurfaces colored in green can be used to produce three-image multiplexing because they belong to the intersections  $R_1, R_2, R_3, R_4, R_5, R_6, R_7$  and  $R_8$ .

### Extension to $N$ modes

This conditional architecture can extend to more than three modes, following the logical color tree. The minimum number of metasurfaces contained in the selected clusters at each step depends on the number of remaining colors to pick in the remaining modes. If  $i$  is the number of the current mode and two colors are to be selected in each mode, the two selected clusters in the current mode should contain at least  $2^{M-i}$  metasurfaces in their intersection with previous clusters.

### Extension to $K$ colors per mode

Similarly, this conditional architecture can also extend to more than two colors per mode. The minimum number of metasurfaces contained in the selected clusters at each step is increased. If  $K$  is the number of colors to be selected in each mode, and  $i$  the

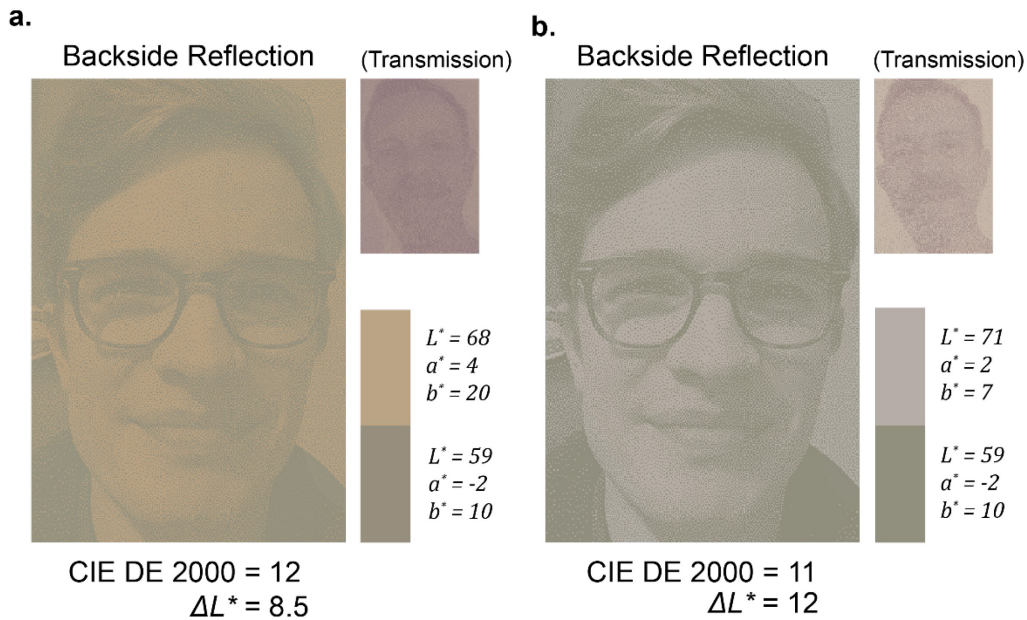


number of the current mode, the  $K$  selected clusters in the current mode should contain at least  $K^{M-i}$  metasurfaces in their intersection with previous clusters.

### ***Selection of the best solutions***

In order to obtain the best image contrast for the multiplexed images, an additional step (step 6 in Algorithm 1 and 7 in Algorithm 2) filters the solutions to keep only those that give both the highest colorimetric distance and the highest contrast (lightness difference) between the two selected colors (clusters) in each mode. The user can then select among the proposed solutions those whose colors better fit his expectations in terms of image rendering.

When only binary images are multiplexed, such as what was done in our experimental implementation, the  $\Delta L^*$  seems to have a greater impact on the image quality. Figure 5.6 illustrates this by showing two simulations of a multiplexed image seen in mode “backside reflection”, constructed with the colors of two solution found by the algorithm on our metasurface database. They both have a similar contrast in terms of CIE DE 2000 distance between the colors of their clusters, but the one having a higher contrast in terms of  $\Delta L^*$  looks better. This correlates with the greater sensitivity of the human visual system to luminance variations for high spatial frequencies, such as those present in an image[82]. We thus filtered the solutions by keeping only those whose  $\Delta L^*$  between the two colors in each mode is at least twice the clustering cutoff distance.

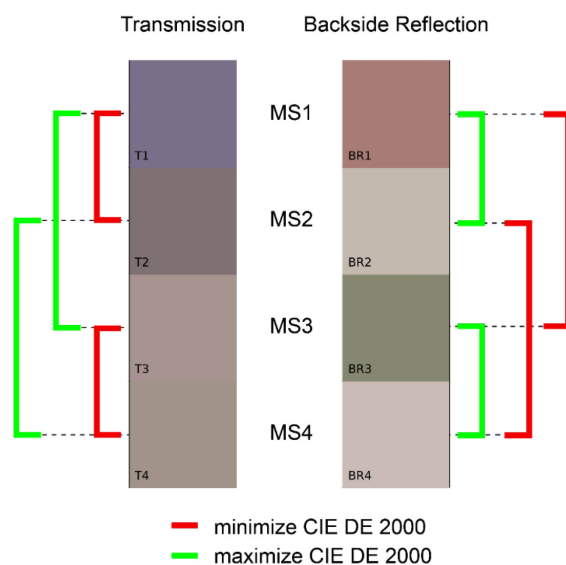


*Figure 5.6 Comparison of two multiplexing solutions which have a similar contrast in mode “backside reflection” in terms of CIE DE 2000 (computed between the two average colors of the clusters displayed in the rectangles next to each backside reflection image) but different contrasts in terms of  $\Delta L^*$ . Note that the backside reflection image of solution (b), which has a higher  $\Delta L^*$  contrast, appears more contrasted. The two solutions were obtained after a clustering on CIE DE 2000 with a cutoff distance of 5.*

When the database consists of a regular sampling of the laser parameters, an additional criterion of selection can be the number of metasurfaces per selected hypercolor. Indeed, if several metasurfaces with slightly varying parameters create the same hypercolor, it indicates their color is more stable with respect to possible repeatability issues. Thus, the solutions for which each hypercolor are created by several metasurfaces is preferred, if existing.

### ***Selection of the best metasurface combination***

Once the solution with the best colors have been selected, the metasurfaces can be retrieved by a simple indexing operation. When several metasurfaces are available for each hypercolor, an additional step chooses the combination which maximizes contrast, while minimizing the color difference between metasurfaces that should display the same color. For this, CIE DE 2000 is used. The goal is to select the metasurfaces combination that minimizes, in each mode, the CIE DE 2000 distance between the actual colors of the metasurfaces that belong to the same cluster, while maximizing the distances between the colors of the metasurfaces that belong to different clusters. These relations are illustrated on the example of a solution with 2 modes and 2 colors per mode in Figure 5.7.



*Figure 5.7 Colorimetric distances to be minimized or maximized between the metasurface colors of a multiplexing solution for 2 colors and 2 modes.*

To perform this metasurface selection, a cost is computed for each metasurface combination, by summing all distances between the colors that should be minimized, and affecting a negative exponential to the distances that should be maximized. If we take the example of Figure 5.7, the cost  $C$  would be:

$$\begin{aligned}
C = & \Delta E_{00}(T1, T2) + \Delta E_{00}(T3, T4) + \Delta E_{00}(BR1, BR3) \\
& + \Delta E_{00}(BR2, BR4) + e^{-\Delta E_{00}(T1, T3)} + e^{-\Delta E_{00}(T1, T4)} \\
& + e^{-\Delta E_{00}(T2, T3)} + e^{-\Delta E_{00}(T2, T4)} + e^{-\Delta E_{00}(BR1, BR2)} \\
& + e^{-\Delta E_{00}(BR1, BR4)} + e^{-\Delta E_{00}(BR2, BR3)} + e^{-\Delta E_{00}(BR2, BR4)}
\end{aligned} \tag{5.2}$$

where  $\Delta E_{00}(T1, T2)$  denotes the CIE DE 2000 distance between the colors of metasurfaces 1 and 2 in mode “transmission” (while BR indicates the colors of mode “backside reflection”).

For each multiplexing solution considered after the filtering, the metasurface combination having the lowest cost is selected in the last step of the algorithm (step 7 in Algorithm 1 and 8 in Algorithm 2).

### ***Constructing the multiplexed image***

The solutions given by the conditional gamut algorithm and subsequently selected after filtering are lists of colors which can be used independently in each mode. Any image with a fixed number of colors can thus be adapted by selecting one of the colors of the multiplexing solution for each color of the original image, thus creating a false color image. In this chapter focused on binary images, the images are simply obtained from grayscale images by error diffusion in black and white, and then converted to the colors of the multiplexing solution in each mode by making sure that the brightest color is selected to reproduce the white and the darkest color is selected to reproduce the black, in the corresponding binary image. Then the metasurface number for each pixel is determined according to the cluster number of the colors in both images.

## **5.4 Solutions given by the algorithm and clustering**

As exposed above, this algorithm can be extended to find a high number of colors and a high number of modes. However, an increase in the number of modes or colors per mode makes the conditions more selective, which might result in an empty set of solutions. A tradeoff must be found between the number of modes and the number of colors per mode, as both ones increase the complexity of the search and thus the risk of not finding any solution. When no solution is found, the desired multiplexing is not possible with the selected database and the selected clustering. However, changing the way the clustering is done can give rise to new solutions.

Changing the clustering mainly means changing the cut-off value, thereby the similarity of the colors within a same cluster. In other words, we can decide that colors which are not strictly identical are similar enough to belong to the same cluster. The image quality is degraded, but can remain acceptable in some cases. For example, in binary images whose two colors are well contrasted (a “pale color”, and a “dark color”, as shown in Figure 5.3), it is not a big issue if the “pale color” is rendered by a set of colors

with slightly different hues. In this case, the clustering distance can be  $\Delta L^*$  instead of the more restrictive CIE DE 2000 distance. Indeed, it has been observed above that luminance contrast is key in the perception of binary images, while the chromatic changes in the color are often ignored by the observer. This releases the constraint on chromaticity, and increases the number of metasurfaces by cluster, thereby the probability of finding metasurfaces in their intersection.

In any case, to increase the number of multiplexing solutions, the clustering cut-off value can be changed. Increasing it in one mode will increase the number of intersections of the cluster of this mode with clusters from other modes. However, as discussed in section 5.2.1, increasing the clustering cutoff distance can lead to the appearance of ghost images. Decreasing it will create more clusters, which can increase the number of solutions when the number of metasurfaces per cluster is high. However, this results in many clusters having similar colors, which will create multiplexing solutions where the images are formed by similar colors, therefore with a very low contrast. A tradeoff has to be found between the number of multiplexing solutions and their quality in terms of uniformity and contrast.

To illustrate this, Figure 5.8 simulates a multiplexed print produced with three multiplexing solutions, observed in mode “backside reflection”. These solutions were obtained by running the algorithm for 2 colors and 2 modes, for different clustering cut-off values in modes “backside reflection” and “transmission” (the same value is used for both modes. The “backside reflection” image is simulated using the colors of the metasurfaces of the solution in this mode. When the cut-off value is low (1 here) only low-contrast solutions are found. Increasing it to 3 allows to obtain a better contrast. Even if small lightness variations appear between metasurfaces of the same cluster, they are not perceptible in the image. On the opposite, when the value is too high (10) a ghost image appears. A subsequent filtering of the solutions can help identify those which have a high enough contrast before the multiplexed print is produced.


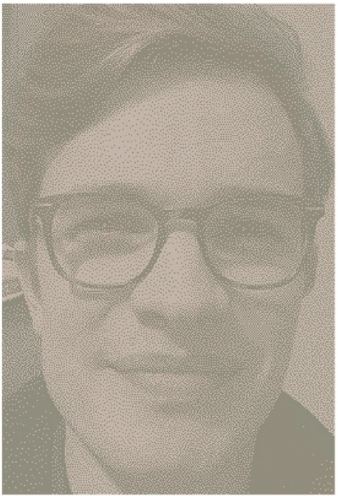
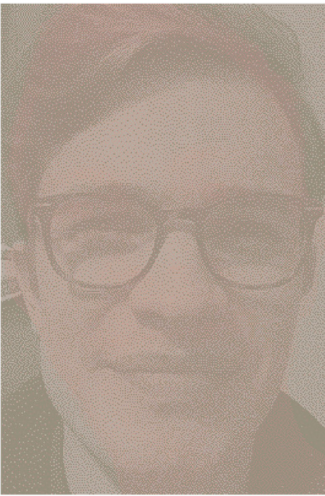
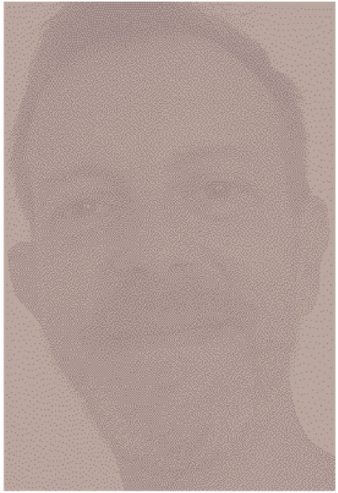
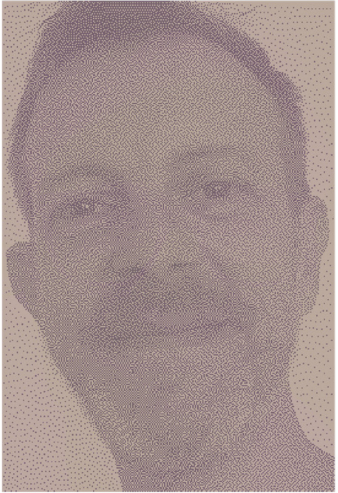

<i>Clustering cut-off (<math>\Delta L^*</math>)</i>	1	3	10
<i>Backside Reflection</i>			
<i>Transmission</i>			

Figure 5.8 Simulation of a demultiplexed image for three multiplexing solutions (2 colors, modes “backside reflection” and “transmission”) found with different clustering cut-off values.

## 5.5 Experimental results

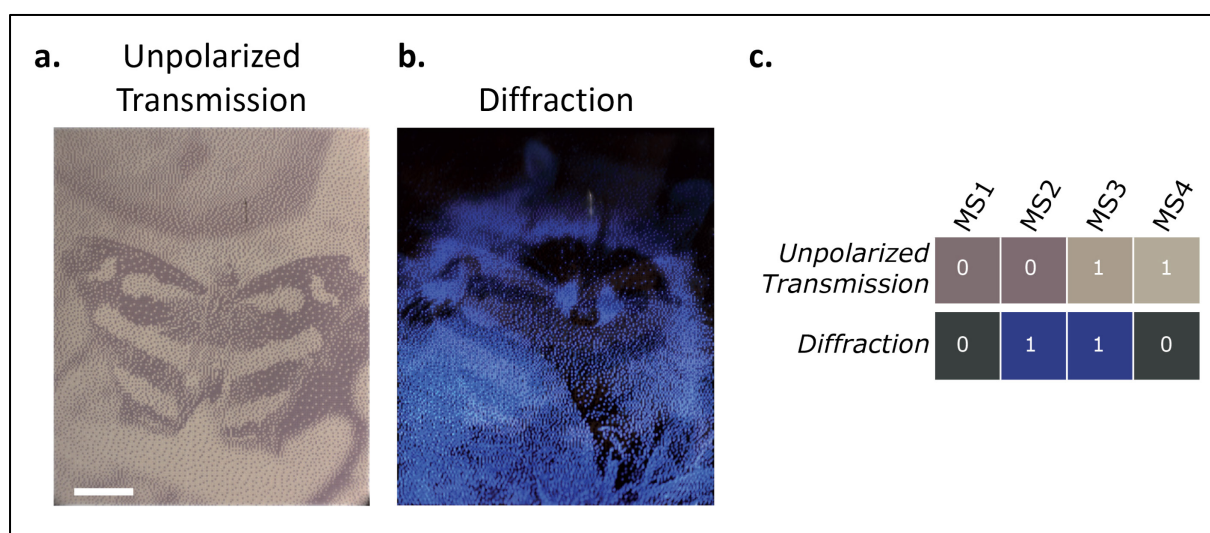
The algorithm was implemented on the sets of RGB colors measured in seven different modes shown in Chapter 3. The following sections demonstrate printed image multiplexing with 2, 3 or 4 images, different combinations of modes and various number of colors per mode, highlighting the potential of combining Ag:TiO<sub>2</sub> thin films and laser-processing for printed image multiplexing.

### 5.5.1 Two- and three-image multiplexing with naked eye observation

This section focuses on modes that can be observed by naked eye to demonstrate the possibility to produce overt security features with two- or three-image multiplexing. The following figures gather the color images recorded under white light with our color

measurement setup, of four nanosecond-laser-processed samples for which multiplexing is implemented for different combinations of modes. All the laser processing parameter sets used to print the multiplexed images of this section are given in Appendix A.

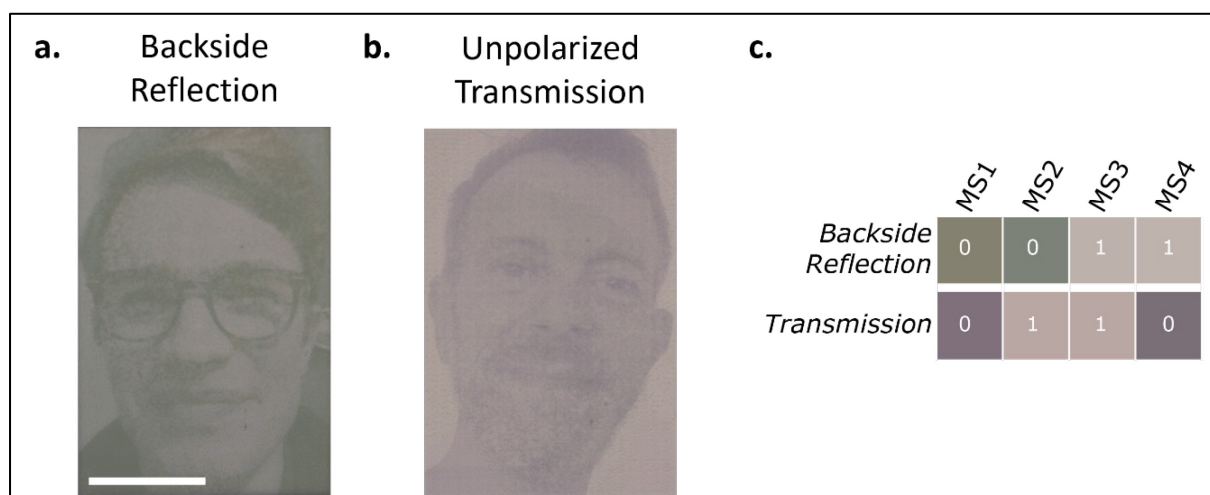
The first demonstration involves “transmission” and “diffraction” modes with pictures of a butterfly and a raccoon (Figure 5.9). The four selected metasurfaces contain self-organized gratings whose orientation is parallel to the incident laser polarization. They are produced with two laser fluences, which give rise to two different colors in transmission and similar colors in the diffraction mode. Each couple (MS1, MS2) or (MS3, MS4) is produced with the same laser fluence. The laser polarization is chosen to create gratings that diffract in the observation plane for MS2 and MS3, and is rotated by  $60^\circ$  to diffract out of the observation plane for MS1 and MS4 as shown in the color tree below the photographs. The true size of the printed images is  $13.4 \times 15.9 \text{ mm}^2$  with a pixel size of  $72 \text{ }\mu\text{m}$ . [Supplementary Movie 1](#)[91] shows the dynamic reading of the multiplexed images with a smartphone camera moving from one mode to the other.



*Figure 5.9 Two-image multiplexing with two colors per mode observed in modes “unpolarized transmission” and “diffraction”. (a) Color photograph of the sample taken in mode “unpolarized transmission”. The length of the scale bar is 2.5 mm. (b) Color photograph of the same sample taken in mode “diffraction” (c) Corresponding logical color tree found by the algorithm.*

The second demonstration relies on modes “backside reflection” and “transmission” with portraits of two members of the research team (Figure 5.10). The image size is  $5.4 \times 8.0 \text{ mm}^2$  with a pixel size of  $20 \text{ }\mu\text{m}$ . The four metasurfaces selected for implementing multiplexing involve parameter sets with various laser fluences, repetition rates and pulse numbers. Their colors in the two modes are arranged in the color tree (Figure 5.10c). [Supplementary Movie 2](#)[92] shows a dynamic reading of the multiplexed images

with a smartphone camera that switches from the “backside reflection” mode to the “transmission” mode.



*Figure 5.10 Two-image multiplexing with two colors per mode observed in modes “backside reflection” and “transmission”. (a) Color photograph of the sample taken in mode “backside reflection”. The length of the scale bar is 2.5 mm. (b) Color photograph of the same sample taken in mode “unpolarized transmission” (c) Corresponding logical color tree found by the algorithm.*

The third sample demonstrates three-image multiplexing observable with naked eye under white light. It involves modes “scattering”, “transmission” and “diffraction” with photographs of three PhD students of our team (Figure 5.11). The image size is  $10.9 \times 15.8 \text{ mm}^2$  with a pixel size of  $72 \mu\text{m}$ . The eight metasurfaces selected for implementing multiplexing involve parameter sets with various laser fluences, speeds, repetition rates, polarizations and interline spacing values. Their colors in the three modes are arranged in the color tree. [Supplementary Movie 3](#)[93] shows a dynamic reading of the three multiplexed images with a smartphone camera.

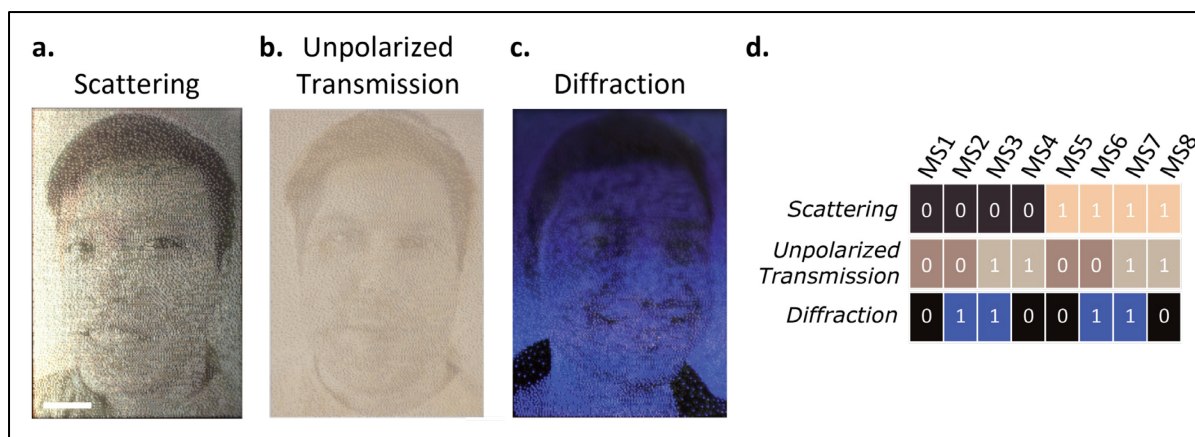


Figure 5.11 Three-image multiplexing with two colors per mode observed in modes “scattering”, “transmission” and “diffraction”. (a) Color photograph of the sample taken in mode “scattering”. The length of the scale bar is 2.5 mm. (b) Color photograph of the same sample taken in mode “unpolarized transmission” (c) Color photograph of the same sample taken in mode “diffraction” (d) Corresponding logical color tree found by the algorithm.

### 5.5.2 Four-image multiplexing using polarized modes

White light three-image multiplexing had already been recently demonstrated using femtosecond laser processing with images displayed in the following modes: reflection with non-polarized light and transmission with the sample placed between two polarizers[54]. It was also demonstrated that with an anisotropic sample, two images can always be multiplexed when using polarized light provided that two azimuthal angles  $\alpha$  of the sample are used to display two images[54]. Here, the methodology is similar as illustrated previously, but the three images are observed in transmission only. One is revealed under unpolarized light, the two other ones being revealed when the sample is placed between two polarizers with two different rotation angles of the sample in its plane. Furthermore, a blank image, hiding the three images, appears in “frontside reflection” mode.



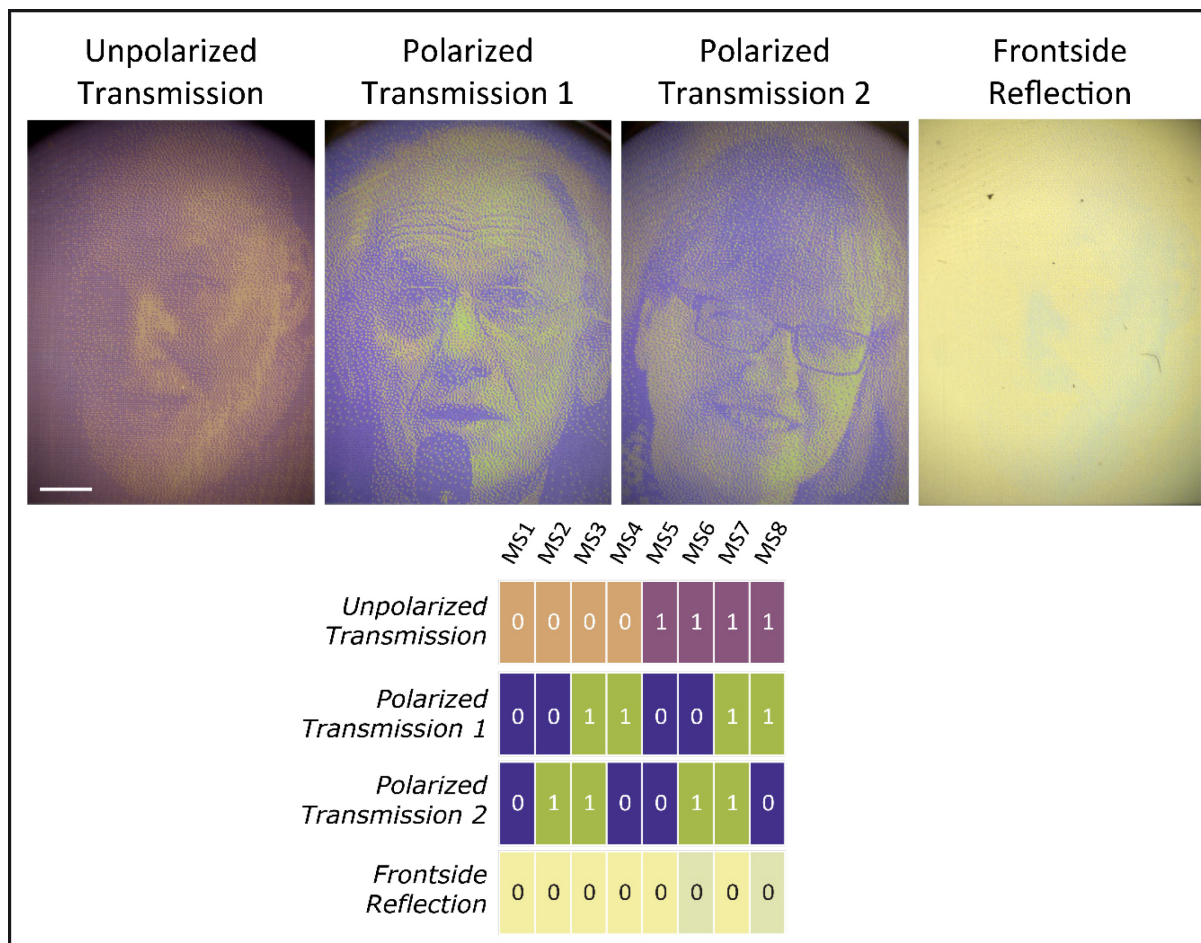


Figure 5.12 Four-image multiplexing with polarized white light. Three images and a blank image are multiplexed to be revealed in “unpolarized transmission”, “polarized transmission 1”, “polarized transmission 1” and “frontside reflection”. The pictures are the photographs of the sample taken with our setup in the corresponding modes. The colors of the eight metasurfaces are displayed in the logical color tree. The length of the scale bar is 2.5 mm.

Apart the blank image, bicolor images are used for this four-image multiplexing demonstration. The eight metasurfaces selected for this implementation all exhibit dichroism caused by the generation of self-organized nanoparticle gratings parallel to the laser polarization. Two speeds (50 and 200 mm s<sup>-1</sup>) are used to get two different colors in the non-polarized transmission mode. Four laser polarization orientations (30°, 60°, 120° and 150°) are used to get two color combinations between the two polarizers (at  $\beta = 85^\circ$  from each other) for two specific azimuthal angles  $\alpha$  of the sample. The colors of these eight metasurfaces in four modes are shown in the logical color tree in Figure 6, together with the photographs of the pictures revealed in each mode. They show the three scientists who shared the 2018 Nobel prize in physics for groundbreaking inventions in the field of laser physics. Arthur Ashkin, appears in the non-polarized transmission mode, whereas Donna Strickland and Gérard Mourou appear in the polarized transmission modes. [Supplementary Movie 4](#)[94] shows the morphing that

appears in the polarized transmission mode when rotating the sample in its plane (azimuthal angle  $\alpha$ ) between the two specific angles where each image alternatively appears. Table A.2 gives all the processing parameters for each metasurface.

### 5.5.3 Multiplexed elements added to a color image

In Figure 5.13, seven primaries are selected in mode “backside reflection” to print a color image. As detailed in Chapter 4, the colors of the initial image are converted to the experimental ones by gamut mapping and color error diffusion is used for rendering color gradients. The second mode here is the “diffraction” one. Among the seven selected color primaries (clusters of mode “backside reflection”), one is part of a partial solution to multiplexing. Indeed, metasurface MS7 diffracts light, which enables to tune its color in mode “diffraction” using polarization. It is thus duplicated with a different grating direction in MS8, giving the same color in “backside reflection” and a different color in “diffraction”. Logos are inserted in the “backside reflection” colored image in the areas where the MS7/MS8 color is present. In this example, the repetition rate and the interline spacing are fixed, while the fluence, polarization and scanning speed are varied. Further, some colors are produced by repeating the laser processing several times in the same pixel introducing another parameter shown in Appendix A (number of passes). [Supporting Movie 5](#)[95] shows a dynamic reading of the color image and the diffracting logos with a smartphone camera.

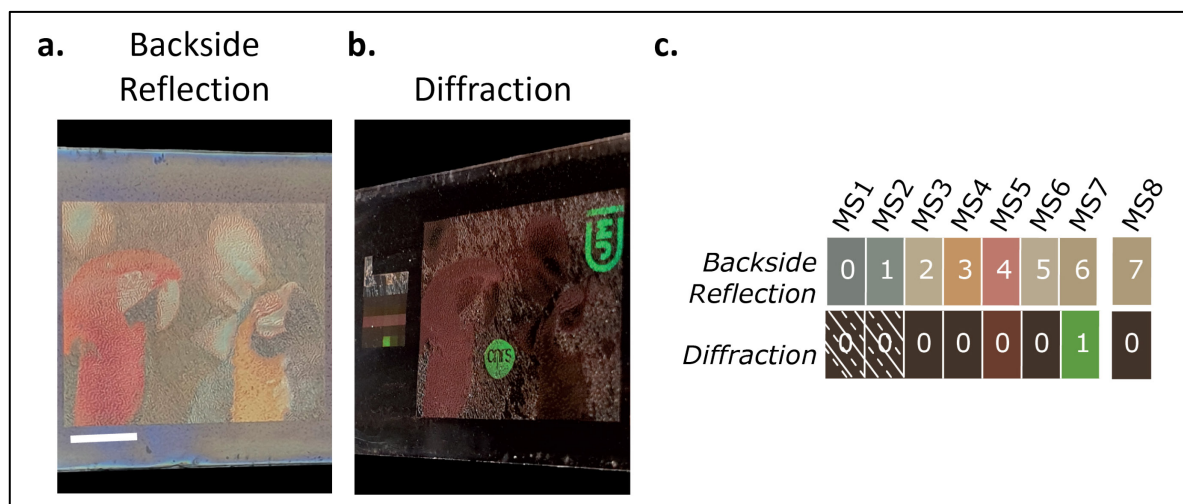


Figure 5.13 Two-image multiplexing with 7 and 2 colors in mode “backside reflection” and “diffraction”, respectively. (a) Color photograph of the sample in mode “backside reflection”. Original image of the two macaws reproduced with permission from the Kodak Photo CD Sampler. The length of the scale bar is 2.5 mm. (d) Color photograph of the same sample taken in mode “diffraction” (d) Corresponding logical color tree where only MS7 and MS8 are multiplexed.

## 5.6 Conclusion

In this chapter, we propose a workflow for multiplexing binary images, using the concept of conditional gamuts. Clustering enables working with colors rather than with metasurfaces, which considerably reduces the complexity of the problem since the number of colors is much lower than the number of metasurfaces in the database. It also provides a first way to explore the performance space of multiplexing. A simple algorithm is described to identify the parameters that allow multiplexing, even when the underlying relation between the design parameters and the produced colors is not known. Using error diffusion enables rendering grayscale images from multiplexing solutions having only two colors per mode.

We show several demonstrators produced with the metasurfaces found by running the algorithm on our metasurfaces database. On top of multiplexing two or three binary images, we show that multiplexing solutions can be adapted to embed multiplexed elements in a color image, or to conceal the multiplexed images in an additional mode. This demonstrates the variety of effects allowed when multiplexing is performed with our nanosecond-laser-processed TiO<sub>2</sub>:Ag metasurfaces. It is also, to our knowledge, the first demonstration of the multiplexing of three images under white light, where no additional observation device is needed. Further, the demonstration of four-image multiplexing proves that various solutions can be found for multiplexing provided that the general methodology described in this chapter is implemented to find them out. The speeds used with this nanosecond laser processing are in average about 50 times higher than the ones reported with the femtosecond laser processing[54] and allowed to produce the presented demonstrators in 5 to 10 min each.

The conditional gamut algorithm is a powerful and simple way of finding multiplexing solutions for a fixed number of colors and modes. The concept of conditional gamut allows to expose the conditional nature of multiplexing, by exploring the available colors in each mode once the previous colors have been selected. By iteratively exploring the modes, this algorithm can be easily generalized to multiplex more images if the technology it is used on permits it. To our knowledge, this is the first general purpose image multiplexing algorithm to have been demonstrated in the literature. With the reduced complexity offered by the clustering step, it can be applied to large databases of objects producing mode-dependent colors to suggest the sets of parameters that can be used for multiplexing images. We hope that this will help identifying other multiplexing-ready technologies and open the way to multiplexing images with a variety of other technologies, even when the multiplexing solutions are not obvious.

However, although this algorithm is efficient for a fixed number of colors and mode, searching for the optimum number of colors and modes with it implies iterating through all combinations of  $K$  and  $M$ . Due to the progression of the algorithm from one mode to the other one, and to the fact that the condition on the number of metasurfaces depends

---

on the chosen number of modes and colors, changing either  $K$  or  $M$  necessitates to start the algorithm again. This is not efficient. To find the optimum number of colors per mode and number of modes for multiplexing with a given database, and potentially identify in which modes some of the multiplexed images can be displayed in colors, another representation of the problem is derived in the next chapter.



## Chapter 6.

# Graph algorithm

The previous chapter focused on the multiplexing of binary images. The conditions for producing multiplexed prints were established based on a logical color tree. The conditional gamut algorithm allowed finding metasurfaces that fulfill these conditions and permitted the multiplexing of up to three images in three modes in a wide variety of modes. This logical color tree representation, as well as the conditional gamut algorithm, can be extended to the multiplexing of more images, with a larger number of colors per mode.

The questions that arise now are: from the metasurface database that we have, can we determine the maximum number of modes that can be used simultaneously to encode a multiplexed image, and which are these modes? In other words, how many images can be multiplexed? How many colors can have these images? To answer these questions, the logical color tree and the conditional gamut algorithm are not sufficient. We introduce in this chapter a new representation, based on the graph theory, which allows a better exploration of the database to determine the highest potential for multiplexing, in particular the highest number of modes and the highest number of colors per mode. This representation highlights the relations between colors in each mode, in an attempt to better understand the limitations that will be imposed to produce multiplexed prints with a given database. A specific kind of graph, called *k-partite cliques*, have been identified to efficiently represent multiplexing solutions in a visually appealing and informative way, even for a high number of colors and modes. Its adequation to our problem is discussed and proven in specific cases. An algorithm to suggest multiplexing solutions based on this graph representation is derived from a recently published work in graph theory[96]. It is able to find multiplexing solutions with a large number of colors. Methods are proposed to select the best ones and use them to produce color images.

### 6.1 Graph representation

A graph is a mathematical structure which models relations between objects. The objects are represented by vertices and the relations between two objects are represented by edges between two vertices. This creates a map in which it is easy to navigate, and from which properties and algorithms can be derived. It is used to represent

systems in computer networks or complex biological systems such as neural connections. Graphs can also be used as the basis of mathematical proofs such as with the four color theorem, which states that any given map can be colored with four colors so that no two adjacent regions of the map is colored in the same color[97]. They are used to derive efficient algorithms in domains such as operations research[98] or image processing[99].

Graphs are promising to represent the relation between the colors of the metasurfaces in each mode. To be able to apply them to multiplexing, a few definitions are given in the following sections and different approaches to construct a hypercolor graph representation will be discussed, as well as how it impacts the search for solutions to 2- and 3-mode multiplexing and beyond.

### 6.1.1 Definitions

The usual mathematical notation[100] of a graph is  $G = (V, E)$  where  $V$  is the set of vertices of the graph and  $E$  the set of edges. Each edge  $e$  is described by a couple of vertices  $u$  and  $v$ :

$$\forall e \in E, \exists (u, v) \in V^2, e = (u, v) \quad (6.1)$$

where  $V^2$  denotes the cartesian product of  $V$ , meaning that both vertices  $u$  and  $v$  are in  $V$ .

Many recognizable graphs can be identified, and they can be used to identify specific kinds of relations in the data. We define here the ones that will be used in the following.

A graph is said *complete* if it contains all possible edges between all the vertices, i.e. if all vertices are adjacent two-by-two. It thus represents data which are tightly connected.

A graph can contain a subset of vertices and edges that form a complete graph, this complete subset is called a *clique*. A subset of a complete graph is a clique. A subset of a clique is also a clique.

Conversely, a subset of a graph which has no edges between its vertices (but possibly edges with other vertices of the graph out of the subset) is an *independent set*.

A graph with  $k$  independent sets is called a *k-partite* graph, and each of its independent sets is called a *partite set*. 2-partite graphs are also called *bipartite*, and 3-partite graphs, *tripartite*.

A *k-partite* graph which contains all possible edges between each pair of vertices that are not in the same partite set, is called a *complete k-partite graph*.

We call *k-partite clique*[101] a subset of a graph, which defines a complete *k-partite* graph. Note that these should not be confused with *k-cliques*, which are cliques which have a number of vertices equal to  $k$ .

$$G \text{ is complete} \Leftrightarrow \forall(u, v) \in V^2, (u, v) \in E \tag{6.2}$$

$$Q \text{ is an independent set of vertices of } G \Leftrightarrow Q \subset V \mid \forall(u, v) \in Q^2, (u, v) \notin E \tag{6.3}$$

$$G \text{ is } k\text{-partite} \Leftrightarrow \exists(V_1, \dots, V_k) \mid \forall i \in \{1, \dots, k\} V_i \subset V \text{ and } \forall(u, v) \in V_i^2, (u, v) \notin E \tag{6.4}$$

Figure 6.1 illustrates some of these recognizable subgraphs. A graph is represented, with vertices named with letters *a* to *i*. The subset of vertices  $\{a, b, c, d\}$  defines a complete graph with 4 vertices. The subgraph induced by these vertices is thus a clique, and more precisely a 4-clique. Note that all subsets of  $\{a, b, c, d\}$  are also cliques, such as all the connected pairs of vertices of the graph. Independent sets of the graph have been colored in different colors. The subset  $\{e, f, g, h, i\}$  defines a 2-partite (aka bipartite) clique. Here also, several bipartite cliques could be identified in its subsets ( $\{h, e, f\}$ ,  $\{i, e, f\}$  etc.).

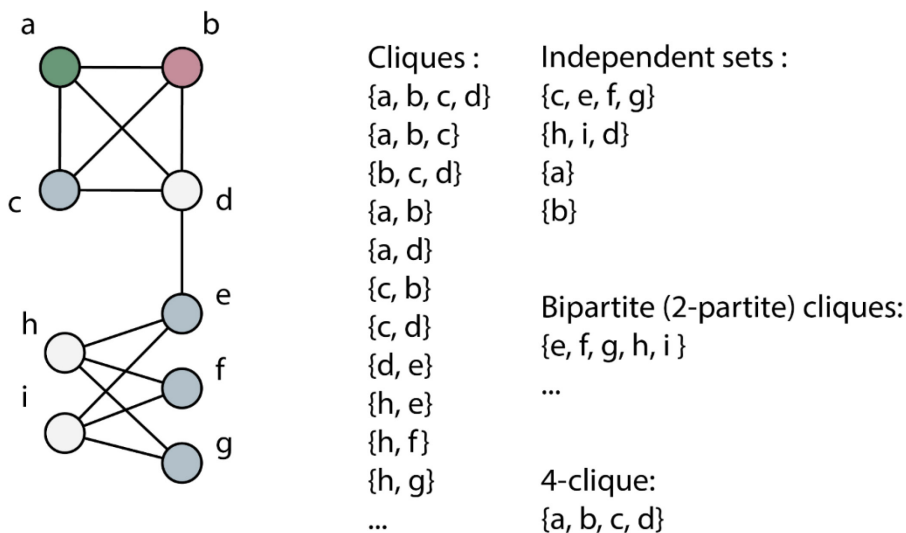


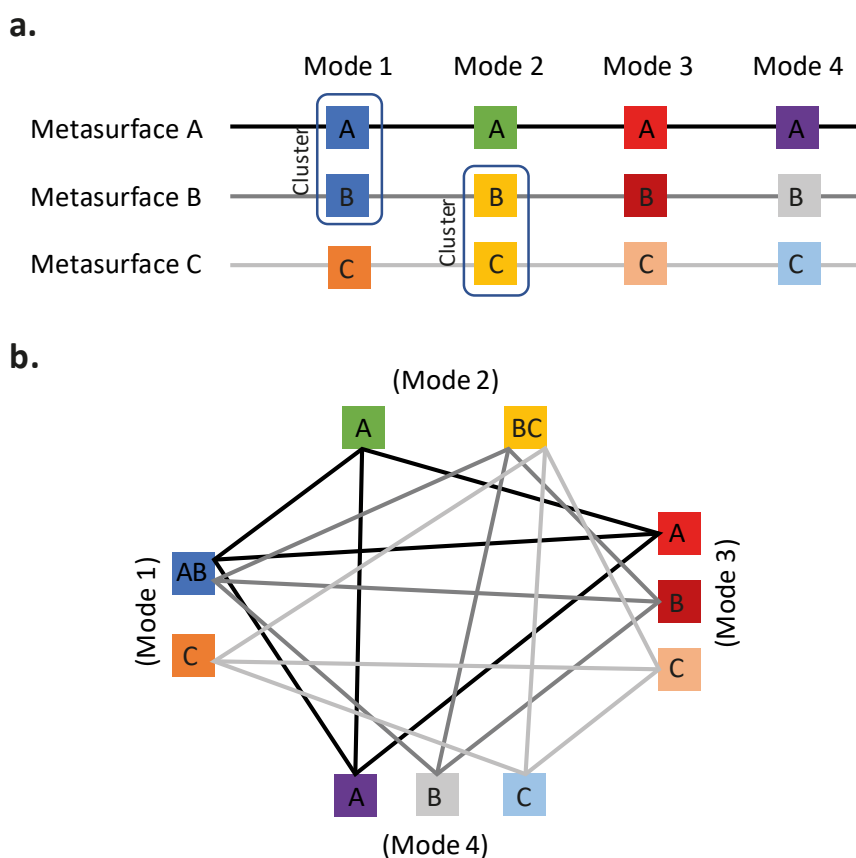
Figure 6.1 Some recognizable subgraphs in a graph.

### 6.1.2 A simple graph representation: 2-mode multiplexing

To represent a hypercolor database, a graph can be constructed after the hierarchical clustering step described in the previous chapter. We decide initially the number of modes *M* that we want to include in the graph. Each cluster is represented by a vertex in the graph, which therefore contains as many vertices as the total number of clusters obtained in all the considered modes. Since each metasurface displays a certain color in each mode, it belongs to *M* clusters, one per mode, and the corresponding clusters are connected by edges. But since one metasurface cannot display more than one color in the same mode, clusters in the same mode cannot contain the same metasurface, and the



corresponding vertices cannot be connected. Therefore, the constructed graph is M-partite.



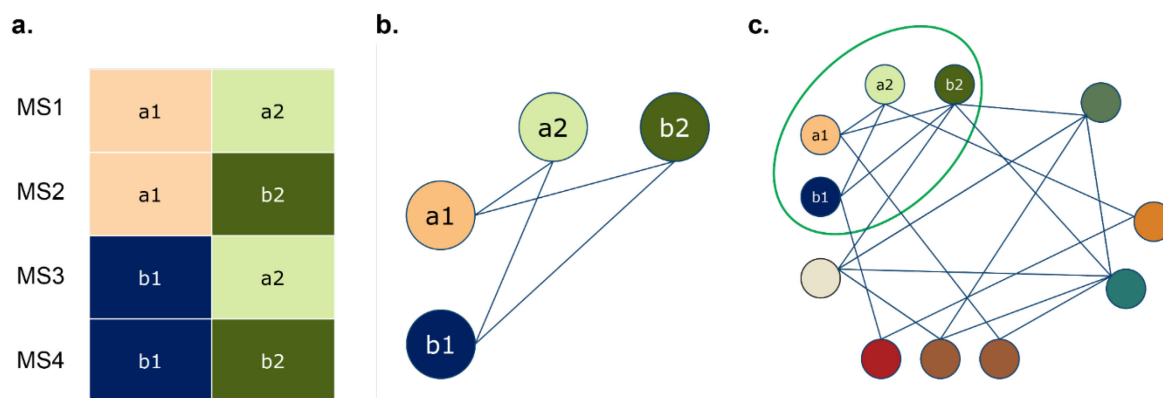
*Figure 6.2 Definition of a graph from a hypercolor database. The hypercolor database (a) contains 4 modes, with 2 colors in mode 1, 2 colors in mode 2, 3 colors in mode 3 and 3 colors in mode 4. The corresponding graph (b) is constructed as follows: One vertex = one cluster; two vertices are linked by an edge when they contain the same metasurface; the clusters of one mode form a partite set since one metasurface belongs to one and only one cluster per mode.*

This graph representation can be used to represent both the full hypercolor database, constructed by thousands of metasurfaces, or single multiplexing solutions. Constructing it from known multiplexing solutions allows to derive a few interesting properties of their graph, which will be used to find them in the full database. In this section, we start by analyzing the graphs of a bicolor multiplexing solution in 2 modes, and discuss its generalization to more colors and more modes.

### ***A 2-mode multiplexing solution is a 2-partite clique***

Figure 6.3 shows the logical color tree and the corresponding graph representing a 2-mode multiplexing solution with two colors in each mode. In mode 1, metasurfaces MS1 and MS2 have been clustered in the same cluster a1, while MS3 and MS4 have been clustered in cluster b1. In mode 2, MS1 and MS3 are clustered together in a2, while MS2

and MS4 are clustered in b2. There are 4 vertices in the corresponding graph, one per cluster : a1, b1, a2, b2. To construct the edges, one lists the clusters couples corresponding to each metasurface in the logical color tree and creates the corresponding edges: (a1, a2), (a1, b2), (b1, a2), (b1, b2).



*Figure 6.3 Hypercolor database (a) and graph representation (b) of a solution for multiplexing with two modes and two colors per mode. 2 modes lead to a 2-partite graph. As the database only contains metasurfaces that form a solution for multiplexing, the graph is a complete 2-partite graph. (c) If the previous 2-partite graph is included in a larger one, it becomes a 2-partite clique. Therefore, to find solutions to 2-image multiplexing we need to find 2-partite cliques in a graph constructed from an experimental hypercolor database. Several 2-partite cliques can be found in the database. The number of vertices in the partite sets that belong to each 2-partite clique corresponds to the number of colors that could be used in each mode for multiplexing. The graph approach allows to find all the 2-partite cliques in a 2-partite graph and thus to rapidly identify the maximum number of colors that can be used for multiplexing. This was not so straightforward with the approach described in Chapter 5.*

In the resulting graph, there are edges between all couples of clusters that are not in the same mode, whereas vertices belonging to the same mode are not linked by any edge. The corresponding graph is thus complete 2-partite: it is 2-partite because the two modes create two independent sets in the graph, as mentioned above; it is complete 2-partite because all vertices that do not belong to the same partite set are linked by an edge.

When considered in the full graph of the hypercolor database, this solution is a *2-partite clique*. We can prove this using the notations defined previously, as well as remembering the conditions necessary to multiplexing defined in Chapter 5.

The implication “If a metasurface set is a solution to 2-mode multiplexing, then it is complete 2-partite” is fairly straightforward. The 2-partite property comes from the result of the hierarchical clustering, which ensures that no metasurface can be in two clusters at the same time, hence there is no edge between clusters of the same mode. The complete property remains to be shown. We can for example show it by contradiction:

Let  $G = (V, E)$  be a graph that is *2-partite* but not *complete 2-partite*. Let  $(V_1, V_2)$  be the independent sets.  $G$  is not complete 2-partite implies  $\exists (v_1, v_2) \in V_1 \times V_2, (v_1, v_2) \notin E$ . If we come back to our definition of the graph representation, it means there are no metasurface which belong to cluster  $v_1$  in mode 1 and cluster  $v_2$  in mode 2. Therefore, one cannot obtain color  $v_2$  in mode 2 for a pixel that is colored with color  $v_1$  in mode 1, which means the metasurface set is not a solution to multiplexing.

To demonstrate the equivalence, the proposition “if a hypercolor graph is complete 2-partite, then it is a solution to 2-mode multiplexing” has to be shown. Similarly, we can prove it by contradiction: if it is not a solution to 2-mode multiplexing, it means that at least one metasurface is missing in the logical color tree. If the metasurface is missing, there will be at least one edge missing in the corresponding graph, so the graph is not complete. Besides, the graph is already 2-partite by definition.

We thus have the equivalence “a metasurface set is a solution to 2-mode multiplexing, if and only if it is complete 2-partite”, which enables us to find solutions to 2-mode multiplexing by looking for complete 2-partite subsets (2-partite cliques) of the whole hypercolor graph. Methods and algorithms to look for these subsets are known and will be described in the next section.

### ***Extension to more colors***

Note that the number of clusters in the solution was not mentioned in the demonstration, which implies that this property works for any number of colors. Therefore, to answer the question “How many colors can I multiplex in 2 modes?”, one can look for the 2-partite clique having the maximum number of vertices in each partite set, which will give the multiplexing solution having the maximum number of colors in each mode. Considering the case of two modes in the previous chapter, this is already better: the number of colors is not fixed in the definition of the criteria and can thus be maximized.

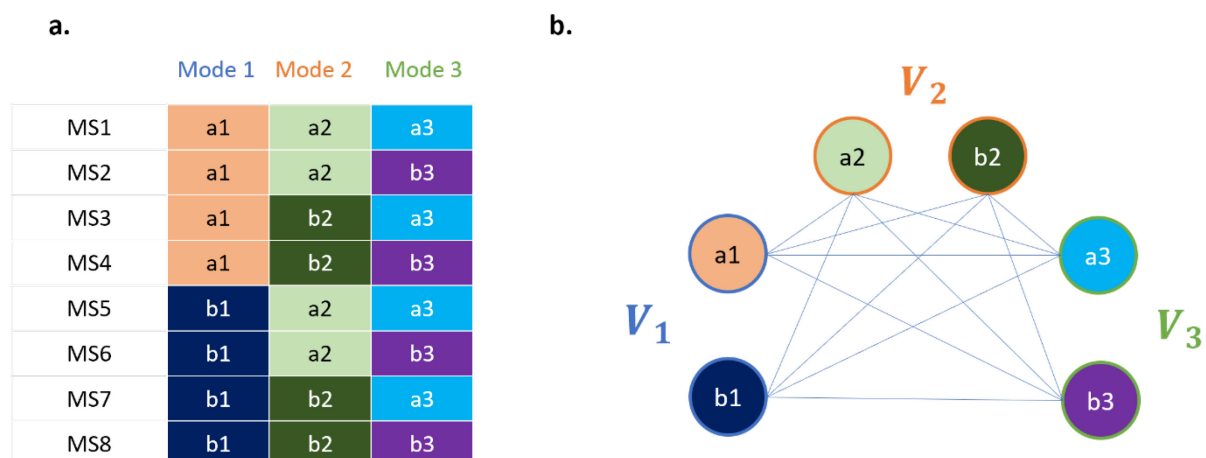
### ***Extension to more modes***

This graph construction enables to identify 2-mode multiplexing solutions among a wide number of modes, by exploring the graph rather than iterating pairs of modes. Indeed, we will see that finding 2-partite cliques in a  $M$  partite graph is well studied and the algorithm which will be described in the following part will function in this case too. However, the extension of this graph construction to find multiplexing solutions in 3 modes and more is not straightforward. It will be discussed in the following section.

### **6.1.3 More advanced graph representations: 3-mode multiplexing and more**

As seen in the previous chapter, a 3-mode multiplexing solution with two colors per mode can be obtained with at least eight metasurfaces. Figure 6.4 shows the graph corresponding to such a solution, with the construction described in the previous section.

This time the constructed graph is a tripartite graph, with partite sets  $V_1$ ,  $V_2$  and  $V_3$  corresponding to the clusters of mode 1, mode 2 and mode 3 respectively.



*Figure 6.4 Hypercolor database (a) and graph representation (b) of a solution for multiplexing with three modes and two colors per mode. 3 modes lead to a 3-partite graph. As the database only contains metasurfaces that form a solution for multiplexing, the graph contains only a 3-partite clique. It is a complete 3-partite graph.*

The affirmation “If a metasurface set is a solution to 3-mode multiplexing, then it is complete tripartite” is still true and easy to show with the same reasoning line as in the case of two modes. Once again, only the complete part of the statement has to be shown because the tripartite part comes from the definition of hierarchical clustering. If one edge misses, it means that one of the colors cannot be obtained if another has been chosen in one of the modes. This is incompatible with the multiplexed images being independent.

However, the equivalence is not true in the 3-mode case. A counterexample is shown in Figure 6.5. Since a 3-mode bicolor multiplexing solution is constructed by at least eight metasurfaces, removing one metasurface from an existing solution with eight metasurfaces necessarily creates a metasurface set that is not a solution to multiplexing, even though this metasurface set creates a complete tripartite graph. Therefore, using the simple graph representation (edges constructed when at least one metasurface belongs to both clusters) a complete tripartite clique does not necessarily yield a 3-mode multiplexing solution.

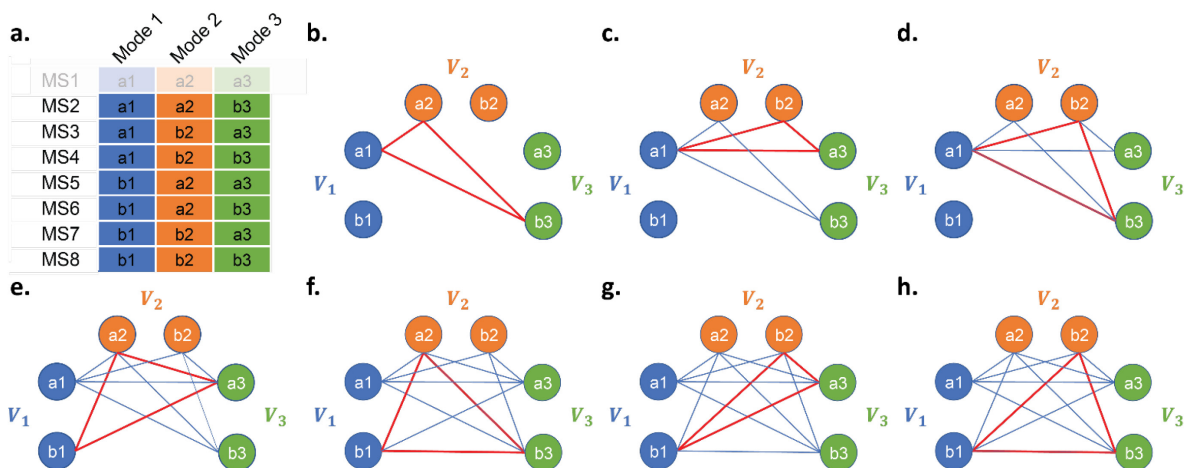


Figure 6.5 Construction of a complete 3-partite graph from an incomplete 3-mode multiplexing solution. a) Logical color tree obtained by removing the first metasurface of a 3-mode multiplexing solution. Construction of the edges created by b) MS2, c) MS3, d) MS4, e) MS5, f) MS6, g) MS7 and h) MS8 (the edges highlighted in red are created by the last considered metasurface).

Another way to see this is to count the number of edges needed to create a tripartite complete graph (Figure 6.6) with 2 vertices per partite set. A tripartite complete graph with vertices per partite set can be obtained by removing 3 edges from a complete graph. By definition, the number of edges in a complete graph is equal to the number of pairs that can be chosen from its 6 vertices, usually noted  $C_2^6$  or “6 choose 2”. Consequently, 12 edges are needed, while up to 3 edges can be constructed by each metasurface. Only four metasurfaces are thus needed to generate such a complete tripartite graph. For example, the metasurface set {MS2, MS3, MS5, MS8} constructs a complete tripartite graph with two vertices by partite set.

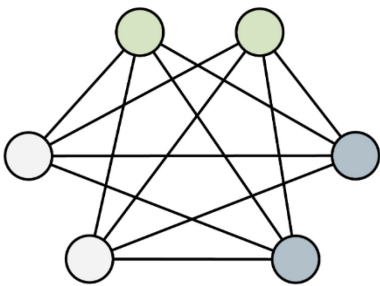


Figure 6.6 A tripartite complete graph with 2 vertices per partite set contains 12 edges. Here, vertices of each partite set are identified by the same color.

With the present construction, an additional step is required to check if the complete tripartite cliques found in the hypercolor graph are really 3-mode multiplexing solutions. To provide a way to directly identify the multiplexing solutions for 3-mode multiplexing

and beyond, alternative graph representations have to be considered. Although less common, they provide interesting ways to tackle this issue at the cost of some additional complexity. We report two possible constructions here and discuss their impact on the implementation of search algorithms.

### **Hypergraphs**

In *hypergraphs*, the (hyper)edges can join any number of vertices[102]. The number of vertices in each hyperedge can vary, however if they all contain  $k$  vertices, the hypergraph is called  $k$ -uniform. A hypergraph can also be  $k$ -partite, when vertices are divided in  $k$  independent sets, and each hyperedge joins exactly one vertex of each set. Manipulating and visualizing hypergraphs is still an ongoing subject of research [103], [104]. Although the similarity with our vocabulary of hypercolors is promising, the absence of out of the box implementations was not compatible with the more experimental nature of our project.

### **Weighted graphs**

In weighted graphs, each edge is attributed a number which is called the weight of the edge. It is commonly used to define a cost (e.g. in the case of the maximal transport) or a flux. Here it can be used to keep track of the number of metasurfaces that construct one edge. A  $k$ -partite clique is a solution to multiplexing only if its edges are constructed by at least  $\prod_{i \neq m_1, i \neq m_2}^M K_i$  metasurfaces where  $K_i$  is the number of vertices of the  $i^{\text{th}}$  partite set and  $m_1$  and  $m_2$  are the indices of the partite sets linked by the edge.

This can be shown by iterating the number of modes and the number of colors. For the initialization, let us consider the case above of a solution for multiplexing 2 colors in 3 modes (Figure 6.4). Let us select  $a_1$  and  $a_2$ , clusters of the two first mode. In the third mode, we must be able to select two different colors,  $a_3$  and  $b_3$ , so the metasurfaces of the solution must represent the hypercolors  $(a_1, a_2, a_3)$  and  $(a_1, a_2, b_3)$ . The number of metasurfaces constructing the edge  $(a_1, a_2)$  is thus at least 2, the number of colors in mode 3 (Figure 6.7a). The same is true for all other pairs of colors and thus for all other edges. In the case  $n+1$ , 2 colors are multiplexed in 4 modes. If we consider the same edge  $(a_1, a_2)$ , the hypercolors that have to be included in the solutions are all the combinations of  $(a_1, a_2)$  and of the colors of mode 3 and 4. Listing all the combinations of the colors of mode 3 and 4 is equivalent to multiplexing 2 colors in 2 modes : at least four metasurfaces are needed (Figure 6.7b). They must construct the following hypercolors :  $(a_1, a_2, a_3, a_4)$ ,  $(a_1, a_2, a_3, b_4)$ ,  $(a_1, a_2, b_3, a_4)$ ,  $(a_1, a_2, b_3, b_4)$ . Therefore, the number of metasurfaces constructing the edge  $(a_1, a_2)$  is at least 4. Similarly, for a higher number of modes  $M$  and a number of colors per mode  $K_i$  ( $i \in \{1, \dots, M\}$ ), the minimum weight of the edges is the number of metasurfaces needed to multiplex the number of colors in the remaining  $M - 2$  modes. As seen in the previous chapter, multiplexing  $K_i$  colors in  $M - 2$  modes requires  $\prod_1^{M-2} K_i$  metasurfaces, thus the condition.

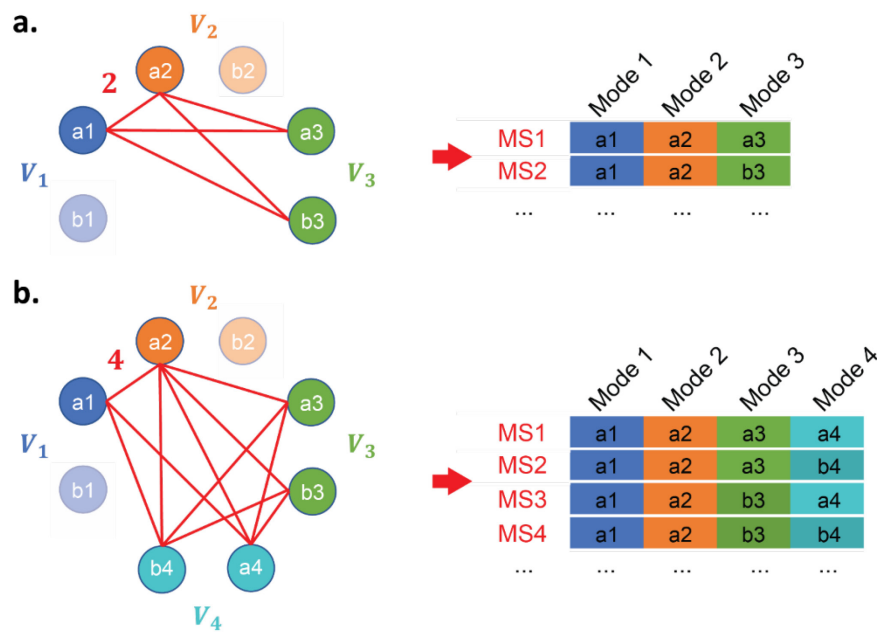


Figure 6.7 Construction of the weighted edge  $(a1, a2)$  for a multiplexing solution with: (a) 2 colors in 3 modes, in which case the minimum weight is 2; (b) 2 colors in 4 modes, in which case the minimum number is 4.

There is however no equivalence between this condition on the weight of the edges and the clique being a solution to multiplexing. Indeed, a set of metasurfaces where two or more metasurfaces constructs the same hypercolor can meet the edge weight condition while lacking one of the required hypercolors for multiplexing.

### Considerations

These alternative graph constructions can provide a representation for multiplexing solutions that will be generalized to three modes, and probably even more with additional conditions. However, they require more complex definitions of the graph, and, at least in the case of the weighted graph, do not yet provide an equivalence between finding a  $k$ -partite clique and finding a  $k$ -mode multiplexing solution when  $k > 2$ . We lacked the time to implement further these alternative graph constructions in this work. However, they are presented here as a perspective to obtain a better representation of the theoretical problem, which might help derive even more powerful algorithms in the future.

In the following, we will work with the simplest graph representation, where two vertices are linked if they have a metasurface in common. Finding a 2-partite clique in this graph directly gives us multiplexing solutions in 2 modes, with a number of colors that is not fixed a priori. Finding a  $k$ -partite clique with  $k > 2$  informs us on the tight connections between the considered set of clusters, but do not give a solution to multiplexing in  $k$ -mode directly.

## 6.2 Algorithms and implementation

Finding  $k$ -partite cliques is part of a more general and well-known problem in graph theory: the *clique problem*, which was stated as one of the 21 problems cited in the article defining the *nondeterministic polynomial-time complete* (NP-complete) class, a class of computationally hard problems in computer science[105]. However, the identification of specific properties of the hypercolor graph reduces the complexity of the problem in our case.

### 6.2.1 Formulation of the problem

The clique problem can be formulated as follows : “Given a graph  $G$  and a fixed integer  $k$ , does  $G$  contain a clique of size  $k$  ?”. This formulation is called the *clique decision problem*. It is the formulation that was shown to be NP-complete by Karp[105]. Other formulations of the problem are obtained by specifying a condition on the size of the clique. Maximal cliques are cliques to which no other vertices can be added. A maximum clique is a maximal clique whose size is maximum. There can be multiple maximum cliques having the same size. Finding one maximal clique can be done in linear time, but listing all maximal clique can require exponential time[106].

In  $k$ -partite graphs, the clique problem is adapted into the problem of finding  $k$ -partite cliques. The same conditions on the size of the  $k$ -partite cliques can be formulated: maximal  $k$ -partite cliques are cliques to which no other vertices can be added, maximum  $k$ -partite cliques are maximal  $k$ -partite cliques whose size is maximum. Two types of maximum  $k$ -partite cliques can be identified[101] : vertex-maximum  $k$ -partite cliques, that have a maximum number of vertices, and edge-maximum  $k$ -partite cliques, that have a maximum number of edges. Vertex-maximum cliques can be found in polynomial time [107], whereas finding edge-maximum bicliques is NP-hard[108]. In tripartite graphs, both are NP-hard[101]. Figure 6.8 shows examples of solutions to the maximum  $k$ -partite clique problem on examples of  $k$ -partite graphs for  $k=2$ .

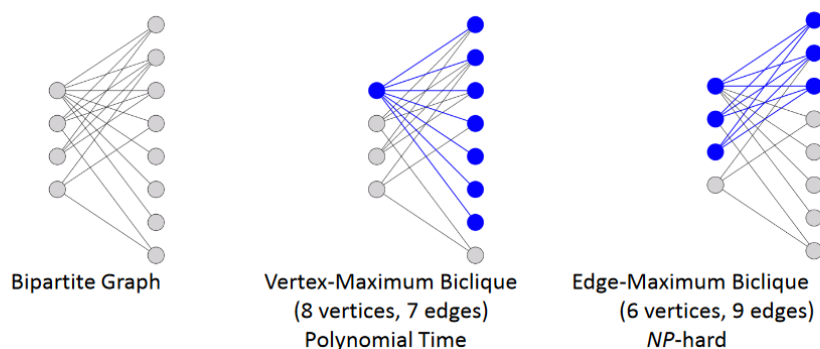


Figure 6.8 Examples of vertex and edge maximum 2-partite clique, reproduced from [109].



In the case of multiplexing, both vertex-maximum and edge-maximum 2-partite cliques are of interest. Searching for the maximum number of vertices is equivalent to maximizing the total number of colors in the solution. It can be of interest when we want to have a color image in one mode, but only a binary image in the other mode. Conversely, searching for the maximum number of edges ensures a more balanced solution with a similar number of colors in each mode. Both are of interest for multiplexing.

However, for multiplexing, other criteria than the number of vertices or edges (e.g. contrast or color gamut volume) should be considered. We also need 2-partite cliques with at least two vertices per partite set. There is therefore an interest in maximal cliques that are not maximum, as they might provide other interesting color palettes, or allow for faster processing speed. For this reason, the search should not only be focused on finding maximum 2-partite cliques, but also on providing a list of maximal 2-partite cliques. Moreover, maximal 2-partite cliques can subsequently be decomposed in smaller 2-partite cliques, which provide 2-mode multiplexing as well, with reduced palettes.

Listing all maximal cliques thus provides several design possibilities. This list will include both vertex-maximum and edge-maximum 2-partite cliques, and enables a subsequent filtering based on the colors, contrast or other criteria relating to the technology (such as marking speed or repeatability). The next section focuses on an algorithm to perform this listing.

### 6.2.2 MMCE algorithm

The Bron-Kerbosch algorithm [110] lists all maximal cliques in a graph by using a recursive backtracking approach, which explores each vertex of the graph in a recursive manner, progressively discarding the vertices that do not contribute to the clique being investigated (backtracking). It was shown to be optimal in the worst case, with a complexity of  $\mathcal{O}(3^{n/3})$  [111]. It is reported to outperform other algorithms [111]–[113]. Based on this algorithm, Philips et al. [96] proposes an algorithm for listing all maximal  $k$ -partite cliques of a graph which is already known to be  $k$ -partite (Algorithm 3). They also suggest a smart way of checking that the outputted cliques are  $k$ -partite, which can be modified easily to control the minimum number of vertices in each partite set, to allow for finding  $p$ -partite cliques in a  $k$ -partite graph with  $p < k$  or to find  $k$ -partite cliques with a predefined number of vertices.

*Algorithm 3 MMCE, reproduced from [96]*


---

```

1  input: a  $k$ -partite graph  $G = (V, E)$ , with partite sets  $V_1, V_2, \dots, V_k$ ;
2  output: all maximal  $k$ -partite cliques in  $G$ ;
3  add all possible intrapartite edges to  $G$ ;
4   $R \leftarrow \emptyset; P \leftarrow V; X \leftarrow \emptyset$ ;
5  ENUMERATE ( $G, R, P, X$ );
6  end MMCE
Subroutine ENUMERATE ( $G, R, P, X$ )
1  input: a graph  $G = (V, E)$ , with vertex partition  $V_1, V_2, \dots, V_k$ , a clique  $R$  that covers this partition, and
two disjoint subsets  $P$  and  $X$  such that  $P \cup X = \{v \in V; R \subseteq N(v)\}$ ;
2  output: all maximal cliques covering this partition that extend  $R$  with vertices in  $P$ ;
3  if  $P = \emptyset$  and  $X = \emptyset$ 
4      then if  $R$  covers the partition  $V_1, V_2, \dots, V_k$ 
5          then report  $R$  as a maximal  $k$ -partite clique;
6      return;
7  choose a pivot vertex  $u$  in  $P \cup X$  that maximizes  $|P \cap N(u)|$ ;
8  for each vertex  $v$  in  $P \setminus N(u)$ 
9      ENUMERATE ( $G, R \cup v, P \cap N(v), X \cap N(v)$ );
10      $P \leftarrow P \setminus v$ ;
11      $X \leftarrow X \cup v$ ;
12 end ENUMERATE

```

---

**6.2.3 Implementation**

We implemented the MMCE algorithm for multiplexing using Python and the NetworkX package [114]. The hypercolor graph is constructed after the clustering step : for each cluster in each mode, we create a vertex labeled with the name of the mode and the cluster number. The edges are constructed by iterating on the cluster numbers of each metasurface, constructing edges that are not yet present in the graph (Algorithm 2).

As exposed above, this forms a  $M$ -partite graph, whose partite sets can be identified in NetworkX by labelling the vertices with a “mode” attribute. Our implementation of the MMCE algorithm is derived from NetworkX’s implementation of the Bron-Kerbosch algorithm, modified as suggested by the authors of [96] to find solutions that contain at least 2 colors in at least 2 of the modes.

---

*Algorithm 4 Hypercolor Edges Construction*

- 1 Input: metasurface table  $T$  such that  $\forall i \in \{1, \dots, N\}$  and  $\forall j \in \{1, \dots, M\}, T(i, j)$  is the cluster number of metasurface  $i$  in mode  $j$ ,  
vertices  $V$  of the hypercolor graph
  - 2 Output: edges  $E$  of the hypercolor graph  $G = (V, E)$
  - 3  $E \leftarrow \emptyset$ ;
  - 4 for  $i$  from 1 to  $N$
  - 5     for  $j$  from 1 to  $M - 1$
  - 6         for  $k$  from  $j+1$  to  $M$
  - 7             if edge  $\left( (j, T(i, j)), (k, T(i, k)) \right)$  does not exist then
  - 8                  $E \leftarrow \left( (j, T(i, j)), (j, T(i, k)) \right)$ ;
  - 9 end Hypercolor edges construction
- 

### 6.2.4 Selecting solutions and retrieving the metasurfaces

Once the 2-partite cliques have been listed and the ones which do not hold the right number of metasurfaces discarded, the algorithm provides a list of possible multiplexing solutions, indicating which colors can be selected in each mode. Within the tolerance of the clustering cutoff, these colors can be multiplexed without generating ghost images, thus allowing to prepare the images in each mode independently. As with the conditional gamut algorithm presented in the previous chapter, the solutions have to be filtered according to the image quality we can expect from them, and the metasurfaces encoding each of the hypercolors have to be retrieved from the clustering table. We suggest a way of filtering the solutions based on the contrast and color gamut they provide. The way to construct the multiplexed image from the color palette available in each mode is also discussed.

#### *Filtering of multicolor multiplexing solutions*

The number of colors in each mode given by the solutions found with this graph approach can be greater or equal to two. For the modes where the number of available colors is two, the quality of the solution can be assessed in the same way as for the results of the conditional gamut algorithm with binary images. When the number of colors is greater, selecting the best solution can be done according to two different objectives: on the one hand, it has to ensure that the contrast in the image is sufficient, which can be done in the same way as for binary images (by using the  $\Delta L$  \* metric) to convey the spatial information present in the images; on the other hand, the color variations in the images can be reproduced by using the colors of the solution as primaries for halftoning. To select the solutions which will give the best color reproduction in this case, the color gamut accessible with each solution in each mode must also be evaluated.

One way of evaluating the gamut accessible with halftoning is to compute the convex hull of the colors in a color space that is linear with respect to halftoning. As the contrast

in luminance can already be assessed by computing the maximum  $\Delta L^*$  between the colors having respectively the highest lightness and lowest lightness, we compute the convex hull in the CIE 1931 xy chromaticity diagram. To obtain a quality measure that is comparable to the  $\Delta L$ , defined between 0 and 100, we divide this area of the convex hull by the area covered by a reference gamut, sRGB in our case, and multiply it by 100 to match the range of  $\Delta L$ .

If  $C_i = \{c_1, \dots, c_n\}$  are the colors in mode  $i$  for the considered solution and  $A_i$  the area of their convex hull in the xy plane, we can define the quality measure of the solution in mode  $i$  as:

$$Q_i = \max_{(c_j, c_k) \in C_i^2} (\Delta L(c_j, c_k)) + \frac{A_i}{A_{ref}} \times 100 \quad (6.5)$$

where  $A_{ref}$  is the area of the reference gamut in the xy diagram.

We thus have a way of quantifying the quality of a solution in each mode independently. To obtain a global quality measure for the whole solution, we take the geometrical mean of the quality measure in each mode. Indeed, if a solution to multiplexing has a quality of 0 in one mode (for example, only two colors are available in this mode and they do not provide any luminance difference,  $\Delta L^* = 0$ , then the multiplexing will not be good, so the overall quality of the solution should be 0, which is true with the geometrical mean. We thus obtain a global quality measure  $Q$  over the  $M$  modes expressed as:

$$Q = \prod_{i=1}^M Q_i \quad (6.6)$$

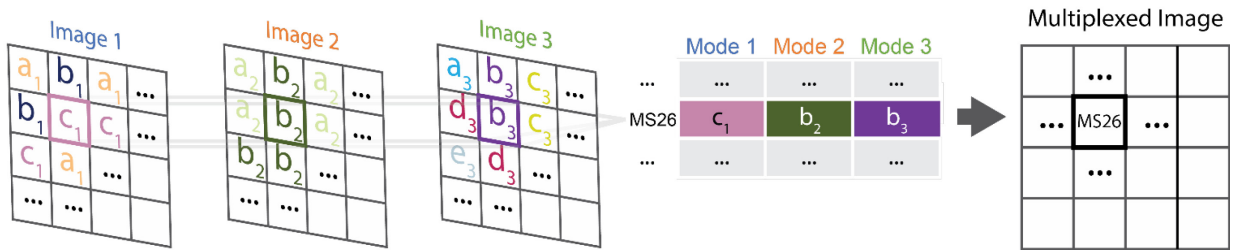
This can be used to rank and filter the solutions.

### ***Constructing multiplexed color images***

Since the colors of the solutions in each mode are independent, the images can be designed in each mode independently. We already covered the case of two colors per image in Chapter 2: the halftoning can be done in black and white, the black and white in the resulting image being given the real colors of the metasurfaces afterwards. We will describe here the case where three or more colors are available in the considered mode. The first steps of the process, gamut mapping and vector error diffusion are performed in a similar way as for a single color image (see Sections 4.1.2 - Gamut mapping and 4.1.3 - Vector error diffusion halftoning). The only difference here is that rather than considering the actual colors of the metasurfaces, the considered colors are the average colors of the clusters selected in the multiplexing solution. This allows to construct the images independently of which metasurfaces of the clusters will be used to produce the

color, as the choice of the metasurfaces will be performed in a subsequent step taking all images into account. The colors of the image are adapted to the gamut defined by the cluster colors by the gamut mapping step. Then vector error diffusion outputs an image where each pixel is attributed a cluster number (and color) from the considered solution. The singular step with this multiplexed color reproduction workflow is that cluster number images are combined and the metasurfaces numbers are retrieved from the series of cluster numbers in each pixel.

Similar to what is done in Chapter 3, the metasurfaces corresponding to the different hypercolors in the logical color tree have to be retrieved in the original metasurface database. Once each multiplexed image has been mapped and halftoned to each available palette, the metasurfaces are retrieved pixel by pixel by considering the cluster number attributed to this pixel in all the modes (Figure 6.9). This is done by a simple indexing operation on the cluster numbers. If several metasurfaces produce the same hypercolor in the considered modes, they can be picked based on parameters preference, reproducibility, or to give the best homogeneity for solid color areas in the image. For this, see the generalization of vector error diffusion in Chapter 7.



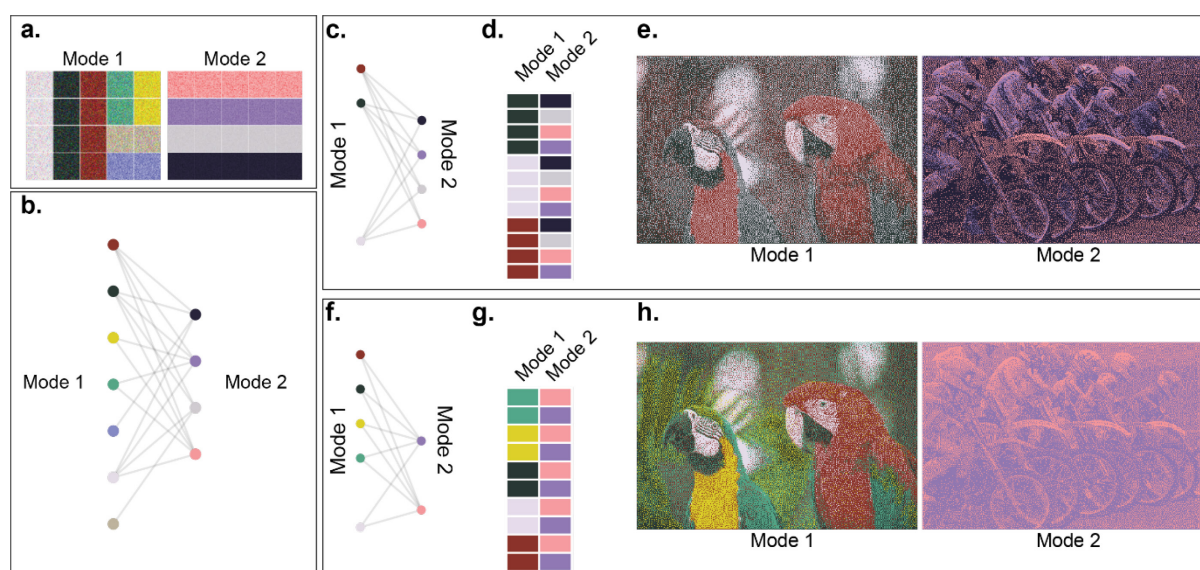
*Figure 6.9 Retrieval of the metasurfaces after construction of the color images. Example with three images multiplexed in three modes. The gamut mapping and clustering steps are performed on each image independently by considering the average colors of the clusters of the solution in the corresponding mode. This results in three images where each pixel is associated to a specific cluster. For each pixel, the three cluster indices are read in the image and used to retrieve the right metasurface selected in the multiplexing solution, and the multiplexed image is constructed.*

### 6.3 Experimental results

The graph algorithm for multiplexing is applied and validated on both synthetic and experimental databases and the provided solutions are evaluated. As pointed out in section 6.1.3, the k-partite representation is well suited to search for 2-mode multiplexing solution and will be applied to this aim here. This demonstrates that the construction of a M-partite graph and the search for k-partite cliques can be performed with our implementation of the MMCE algorithm, and that 2-mode multiplexing solutions with a large number of colors can be found efficiently.

### 6.3.1 2-mode color multiplexing solutions from a synthetic database

The first example showcasing our graph algorithm for multiplexing is obtained by applying it on a synthetic hypercolor database. This database was constructed with homogeneous color patches, to which noise was added to simulate experimental variations in the color due to the laser inscription or to the color measurement process. These virtual colors are used only for this section, to design an easily understandable example of our graph representation and algorithm, they are not extracted from the metasurface database.



*Figure 6.10. 2-mode multiplexing solutions found in a synthetic hypercolor database (a) two images used to create a synthetic hypercolor database. Each patch in each image represents one synthetic metasurface characterized in two modes, mode 1 and mode 2. (b) 2-partite graph constructed from the synthetic hypercolor database.  $K$ -partite cliques (c, f), corresponding logical color tree (d, g) and simulated color images (e, f) for two solutions found in this database : a solution having 3 colors in mode 1 and 4 colors in mode 2 (c, d, e) and a solution having 4 colors in mode 1 and 2 colors in mode 2 (f, g, h). The simulated images have been created by applying our color reproduction workflow independently in each mode to construct each image with the colors of the solution in the corresponding mode.*

The resulting images on which the color measurement is performed is shown in Figure 6.10a. This simulates a hypercolor database which has been characterized in two modes “mode 1” and “mode 2”. This database was designed to give two multiplexing solutions from a reduced number of metasurfaces, to ease the reading of the constructed graph . The hypercolor graph shown in Figure 6.10b is constructed from this synthetic database. Our implementation of the MMCE algorithm is run on this graph and the two multiplexing solutions are found correctly. Figure 6.10c,f shows the two  $k$ -partite cliques found by the algorithm. Figure 6.10d,g shows the two solutions in a logical color tree form

after the synthetic metasurfaces have been retrieved from the database. With these two solutions, gamut mapping and vector error diffusion are performed, and we obtain the two simulated images for each solution, shown in Figure 6.10e,h.

The two solutions found have different advantages and drawbacks: the first solution yields a good contrast in mode 2, but the lack of green and yellow in the palette gives a low color fidelity in mode 1. The second solution, however, presents a better color reproduction in mode 1, but the contrast is lower in mode 2.

### 6.3.2 2-mode color multiplexing solutions from our metasurface database

This second example is obtained by constructing the graph from our metasurface database. To keep the graph readable, only a subset of the database has been used to create this example, with only four modes: “unpolarized transmission”, “frontside reflection”, “backside reflection” and “diffraction”. Figure 6.11 shows the constructed graph, where each vertex is colored with the average color of the corresponding cluster, and edges are semi-transparent black lines. To allow for solutions enabling color images in mode “backside reflection” clustering was performed with CIE DE 2000 and a low clustering cutoff distance of 3. In modes “unpolarized transmission”, “frontside reflection”, and “diffraction”, only luminance contrasts were considered, with a clustering on  $\Delta L$  with a cutoff distance set to 7.

The MMCE algorithm is applied on this graph and 4253 k-partite cliques are found. Among them, 2273 multiplexing solutions were identified by retrieving the metasurfaces in the database. Simulations of the multiplexed images produced with some of these solutions are shown in Figure 6.12. Two-image multiplexing with several colors in mode “backside reflection” can be achieved. The contrast and color fidelity varies from one mode pair to the other.

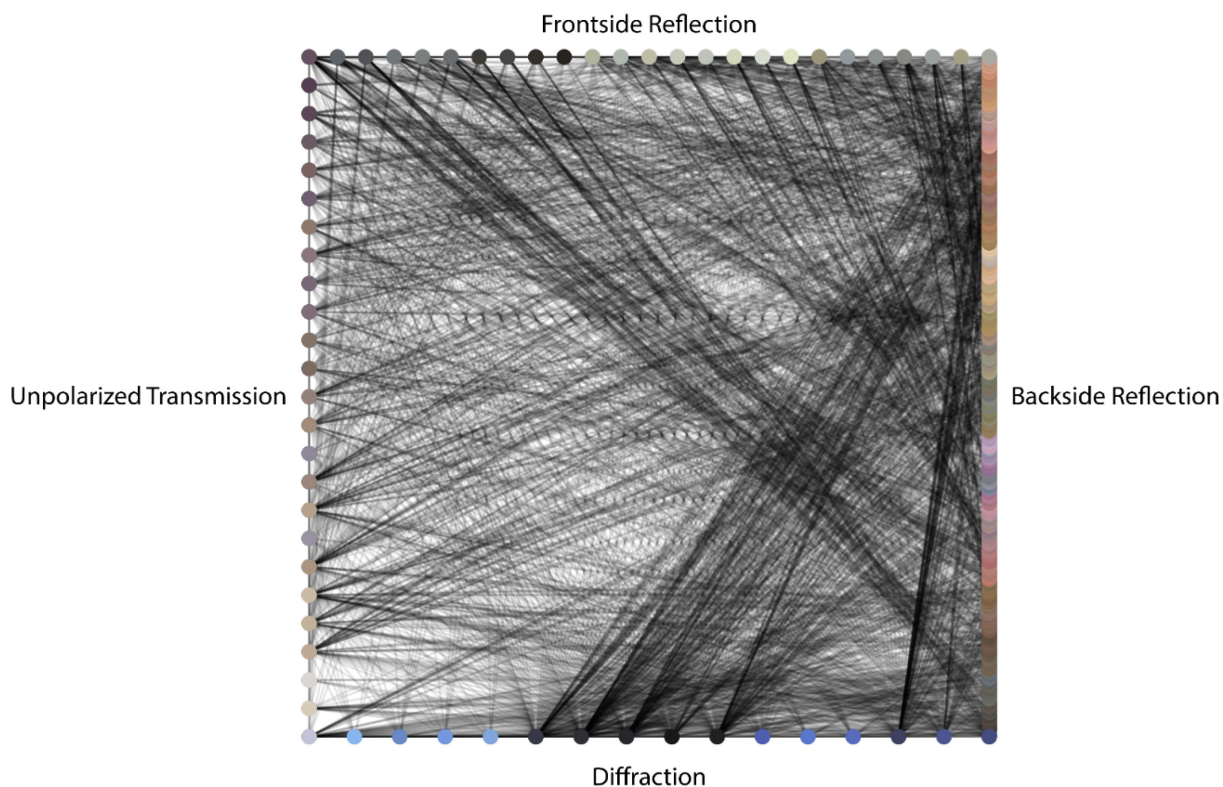


Figure 6.11 Graph constructed with a subset of our database in modes “unpolarized transmission”, “frontside reflection”, “backside reflection” and “diffraction”

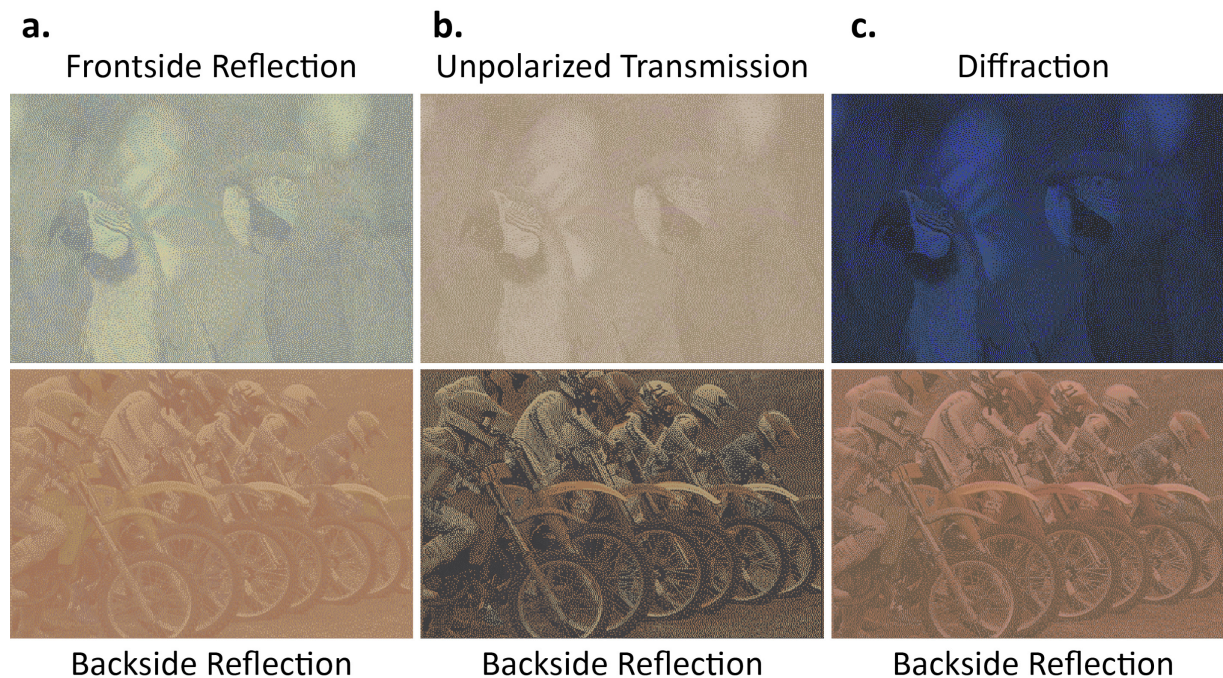


Figure 6.12 Simulated images for three multiplexing solutions found with the graph algorithm in modes “frontside reflection”, “unpolarized transmission”, “diffraction” and “backside reflection”.



Additionally, we found the results to be very dependent on the clustering cutoff distances defined in each mode. In the case of two or three modes, manual exploration was successfully performed by varying the cluster cutoff distances in the modes where multiplexing solution had been found, to optimize the produce images. This optimization was mainly done by trying to maximize the contrast of the best solutions, while visually checking that no ghost images were observed in the simulated image. Here, this manual process is much more difficult as several quality criteria need to be taken into account, in particular the number of colors in the solutions, the presence of solutions in more than two modes, and the gamut volume and contrast of the solutions.

## 6.4 Conclusion

Using the formalism from graph theory, we derived a new representation of the multiplexing problem: a hypercolor graph is constructed by adding a vertex for each cluster in each mode, and by adding an edge between clusters which share at least one hypercolor. It allows to represent the relations between the hypercolors obtained after clustering, for an arbitrary number of colors and modes.

K-partite cliques, a concept often used to identify tight relations between heterogeneous data, have been identified to describe potential multiplexing solutions. They are directly useful when  $k=2$ , as all 2-partite cliques in our graph are 2-mode multiplexing solutions. Using a recently published algorithm, we are able to find and enumerate maximal 2-partite cliques in our hypercolor graph. Each maximal 2-partite clique is a multiplexing solution, which can have a large number of colors. For a larger value of  $k$ , this equivalence is not true. For this case, other graph constructions, namely hypergraphs and weighted graphs, might give a way of working towards an equivalence between the identified objects and solutions to multiplexing, but would require further investigation.

Provided that 2-mode solutions with a large number of color primaries are found, we derived a color reproduction workflow for multiplexing by observing that each color image can be gamut mapped and halftoned independently, if the cluster colors are considered instead of the metasurface color.

Our simple graph construction, and the subsequent multiplexing workflow was successfully applied on a synthetic database. This showed its ability to suggest new multiplexing solutions with a large number of colors in each mode, and to enable the addition of colors in the multiplexed images.

When applied on part of our metasurface database, it suggested some new solutions with more colors in two modes. Simulations of multiplexed prints were realized with these solutions, and show the possibility of multiplexing a color image in mode “backside

reflection” with a grayscale image, which can be shown in modes “frontside reflection”, “unpolarized transmission” or “diffraction”.

Understanding the relations between the colors in each mode is critical to understand when multiplexing is possible with a given database. In this way, the graph representation of hypercolor databases provides a new set of tools derived from graph theory to tackle this high-dimensional problem. K-partite cliques, and the MMCE algorithm are only examples of this, and we believe further work is needed to unlock the full potential of this graph representation.

The application of both the conditional gamut algorithm and the graph algorithm to our hypercolor database showed that clustering cutoff distance has a big impact on the number of solutions found and on the contrast of these solutions. When two or three modes are considered, and only two colors per mode, the manual exploration of several clustering distance is possible. However, when the number of modes and colors is high, the combined effects of the clustering in each mode on the outputted solutions is difficult to predict and optimizing the clustering distance in each mode is much more difficult. Although the ability to discretize our color palette can be useful, for example if an automatic authentication of the multiplexed print by a digital vision system is considered, or even for data storage application, in most cases this is not required for visual observation. In the last chapter of this work, we seek to remove the clustering step, thus approaching the multiplexing problem in a more continuous way.

## Chapter 7.

# Continuous multiplexing

We followed in Chapters 5 and 6 a discrete approach to manage colors based on a clustering step. This approach has the advantage of allowing to explore the hypercolor database in an iterative and potentially interactive way, and to expose the multiplexing conditions with clear examples. However, because the clustering distance has to be optimized separately for each mode and has an unpredictable effect on the multiplexing solutions, it can become a hindrance when trying to multiplex color images, or when considering a large number of modes. In this chapter, we try to overcome that issue by considering directly the real colors of metasurfaces, not the average colors of clusters.

Moreover, we will take benefit of halftoning already introduced in Section 4.1.3, that we used to create color gradients with the very limited number of metasurface colors selected. By juxtaposing pixels of different base colors (that we will call *primaries*), we obtain areas that are perceived at a distance as having an intermediate color, and these intermediate colors can be produced in an almost continuous way by varying the respective surface coverage of the two primaries. This idea, applied only with colors in one mode so far, can be extended to hypercolors and to multiplexed images: for example, we can juxtapose different metasurfaces that give the same color in mode 1 (the result looks uniform) and different colors in mode 2 (multicolor gradients can be displayed). The hypercolors thus mixed by juxtaposition of metasurfaces are actually new hypercolors that can be added to the database, and new multiplexing solutions can be hopefully obtained. Instead of using the methods previously presented with this enriched database, we propose to use directly the vector error diffusion halftoning with hypercolors in multiplexed images.

As for the design of single mode images, vector error diffusion can fall prey to artifacts, if some of the colors are out of the gamut. We thus present a way of estimating the optimal range of colors of each image to maximize image quality. Considering the multiplexing in a higher dimensional space rather than considering each mode independently allows to optimize the contrast of each multiplexed image simultaneously, by generalizing the concept of gamut mapping to a “multi-mode gamut mapping”.

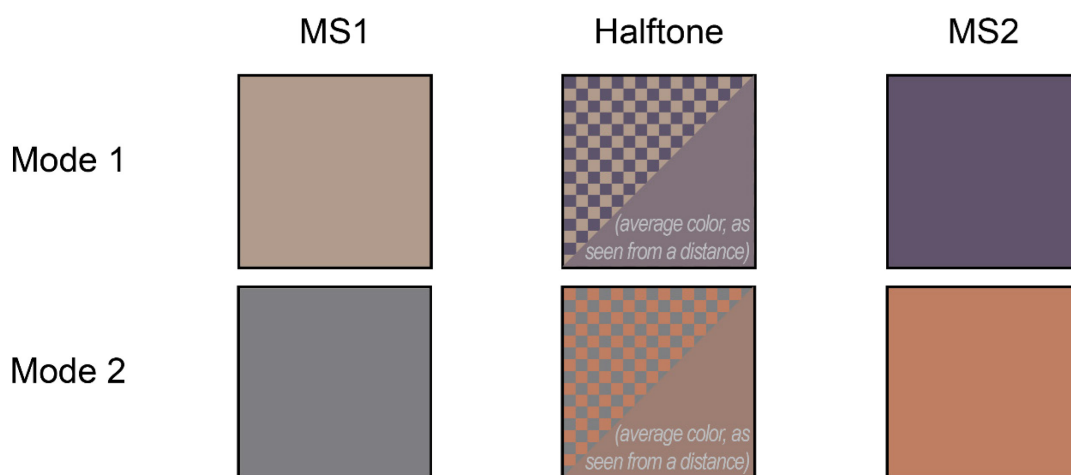
Various strategies are evaluated and implemented to produce multiplexed images with at least one color image and an optimized contrast in all chosen modes. A physical

demonstrator is produced with this approach, showing the multiplexing of a full color image to be observed in mode “backside reflection” and a grayscale image in mode “transmission”.

## 7.1 Multimodal halftoning

As presented in Chapter 2, halftoning relies on the spatial averaging inherent to human vision (or to any imaging device) to render color gradients in a color image. The present chapter relies on its generalization to more modes by making this simple observation: the spatial averaging of colors happens in each mode independently. Therefore, in each mode, the color of a given halftone is the weighted average of its primaries in this mode. We use this simple observation to derive a multi-mode vector error diffusion algorithm that allows the control of the achieved color variation simultaneously in all the modes.

To illustrate how halftoning generates new hypercolors, let us consider two metasurfaces, MS1 and MS2 used as primaries, and a checkerboard-like juxtaposition of these two metasurfaces which correspond to a 50% halftone (Figure 7.1). In each mode, the 50% halftone displays a new color. The halftone therefore gives a new hypercolor.



*Figure 7.1 A 50% halftone based on two metasurfaces A and B displays new colors in modes 1 and 2. It is therefore a new hypercolor.*

### 7.1.1 Extension of vector error diffusion to multiplexed images

Vector error diffusion (VED) can be extended to multiple modes by considering vectors with higher dimension. This is already the case when considering spectral rendering as an objective[115], [116]: instead of a three-dimensional image, the algorithm can take as input a higher dimensional image such as a multispectral or hyperspectral image. The distance used for selecting the primaries in the error diffusion algorithm is extended to a higher dimension.

In the case of multiple modes, the color palette consists in the concatenation of the colors of the metasurfaces in each mode. For example, when there are  $N$  metasurfaces whose color is represented in an RGB system and the number of modes is  $M$ , the resulting dimension of the palette is  $N \times 3M$ . Similarly, the input multiplexed image to be reproduced is a concatenation of the colors of the  $M$  images that are displayed in the  $M$  modes. If the size of the RGB images is  $s_x \times s_y \times 3$ , the size of the resulting multi-mode image is  $s_x \times s_y \times 3M$ .

Once the multi-mode image and palette are prepared, the algorithm follows the same procedure as standard vector error diffusion (Figure 7.2). The multi-mode color of each pixel in the input image is compared to the multi-mode palette, and the best match is selected to reproduce this pixel in the output image. The error is computed as a multi-mode vector and diffused to the neighboring pixels according to the weights defined in Chapter 4. By iteratively processing each pixel of the input multi-mode image, a multiplexed print is constructed.

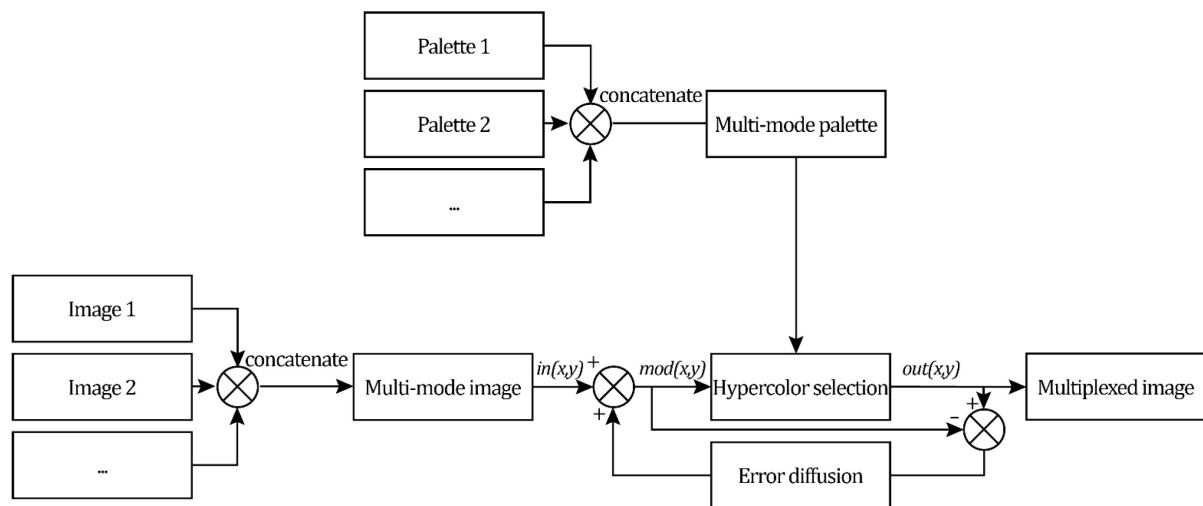


Figure 7.2 Vectorized error diffusion algorithm applied to the multiplexing of two or more images.

### 7.1.2 Necessity of gamut mapping

However, as with single-mode vector error diffusion, multi-mode vector error diffusion is subject to artifacts when the colors are out of gamut. When it encounters colors that cannot be multiplexed, the error generated at the hypercolor selection step can be huge. When only one mode is considered, these artifacts are prevented by the gamut mapping step, which ensures that all colors of the image are mapped to the reproducible gamut before halftoning. The same has to be defined for multiplexed images, otherwise artifacts and ghost images may appear on the multiplexed print. Figure 7.3 shows a result of bi-modal halftoning performed directly with the metasurfaces of our database, in modes “unpolarized transmission” and “backside reflection”. We simulated

the produced multiplexed print and the colors of the metasurfaces chosen in each pixel is retrieved in our experimental database, to display the simulated observed image in each mode. In these simulations, we clearly can see a ghost effect on both images: moto wheels are visible in the birds, and grey areas correlated with the image of the birds are visible on the picture of the mots.



*Figure 7.3 Two simulated images generated by bi-modal halftoning without gamut mapping. The color of each pixel corresponds to the colors of the selected metasurfaces in mode “transmission” (left) and mode “backside reflection” (right). Ghost images appear when colors are out of the multi-mode gamut. Original images are from the Kodak Photo CD Sampler<sup>[15]</sup>.*

## 7.2 Multi-mode gamut mapping: gray scale images

Multi-mode vector error diffusion can provide a way to obtain a continuous color multiplexing while relaxing the constraints discussed in the previous chapters. However, the problem of assessing the available multi-mode gamut still needs to be solved. The interdependence of the gamuts associated with each mode gives to the problem a high dimensionality, and finding a geometrical representation of it in the general case is not an easy task. We propose to first reduce the dimensionality of the problem in order to derive relevant solutions before discussing the general case.

For the sake of introducing continuous tone multiplexing, we start with a virtual example of two grayscale images, and the palettes displayed in Figure 7.4 which have been obtained by converting to grayscale the color images taken by our acquisition setup. Each square is a metasurface characterized by a couple of gray levels, one in each mode. Similar grayscale palettes can also be generated by a custom mixing of the three channels of the RGB image, or by keeping only one of the channels. This example is not only virtual: it could be obtained physically by applying a spectral bandpass filter in front of the camera during measurement or observation of a set of metasurfaces.

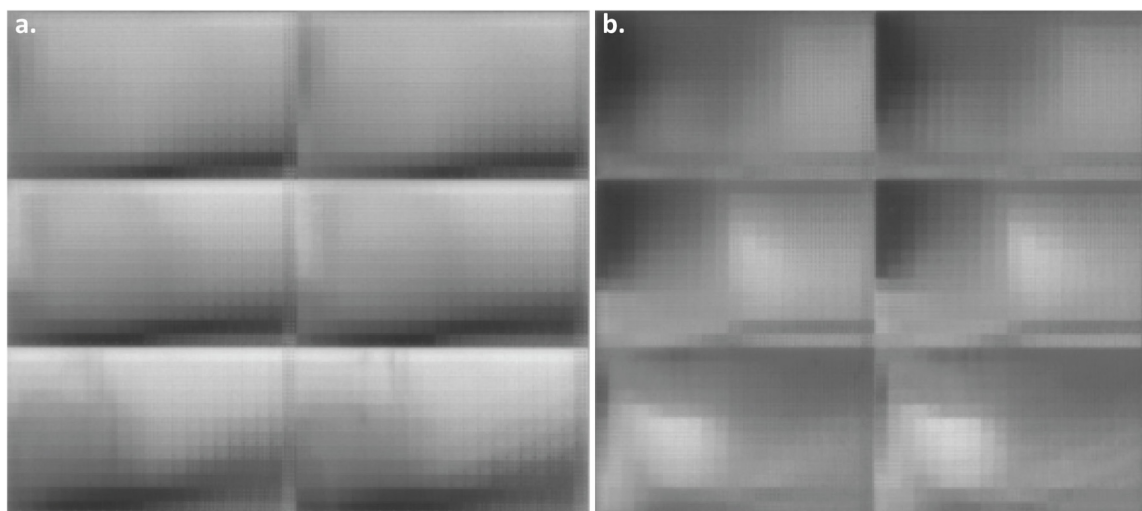


Figure 7.4 Grayscale palettes obtained from 1512 metasurfaces with square pixels taken from our database a) mode “transmission” b) mode “backside reflection”.

To perform continuous multiplexing with these palettes, our task is to identify the parameter ranges for which the images displayed in both modes will be completely independent. If we were to consider the full range of parameters, the gray level chosen in one image would condition the available gray levels available for the other image (conditional gamut), thus creating ghost images and artifacts in the multiplexed image. To identify the ranges where multiplexing is possible, we introduce a 2D representation of this palette, where each mode is represented along one dimension.

The conditionality of the problem is well represented on a two-dimensional plot on which gray levels of mode “transmission” (T),  $L_T$ , are represented on the  $y$ -axis and gray levels of mode “backside reflection” (BR),  $L_{BR}$ , are represented on the  $x$ -axis. Each metasurface of the palette is a point on this plot, colored in blue in Figure 7.5.

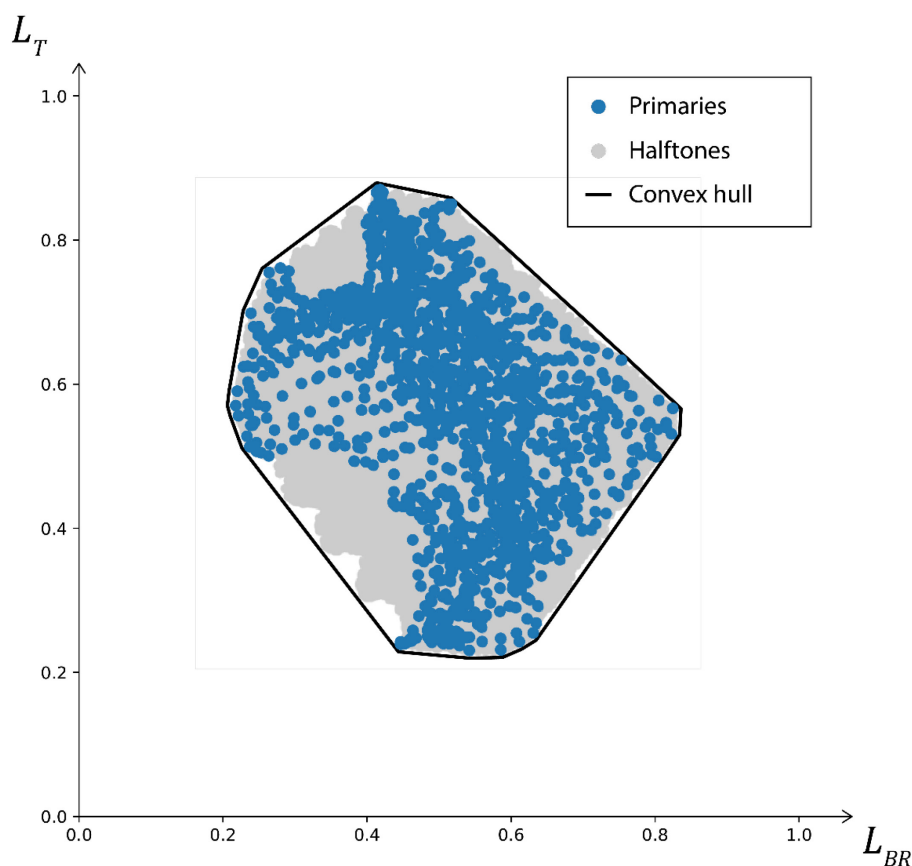


Figure 7.5 2D gray levels multiplexing plot.

In this plot, halftones of two metasurfaces give a new metasurface whose representing point is on the line segment joining their two representing points. This is true for every couple of metasurfaces. Therefore, the range of accessible hypercolors with a palette is its convex hull in this plot, which defines a continuous gamut of hypercolors. In Figure 7.5, 25%-75% and 50% halftones between all couples of metasurfaces of the palette are represented as gray dots. They are indeed in the convex hull of the palette in this multi-mode plot, illustrated by the black line.

### 7.2.1 Multiplexing conditions

The continuous gamut defined by the convex hull in this multi-modal space enables defining continuous multiplexing conditions from a discrete multiplexing palette. In our 2D example, let us consider two gray levels in mode T: 0.4 and 0.6. As shown in the previous chapters, each color (or gray level) selected in mode T gives a conditional gamut in mode BR. The two conditional gamuts can be identified on the graph by following the horizontal lines corresponding to each gray level selected (Figure 7.6). The primaries generating this conditional gamut are the ones located on the line, defined here as an interval of  $\pm 0.01$  around the desired gray level for the sake of this demonstration. The conditional gamut associated to each gray level can be assessed by intersecting this line



with the previously defined convex hull. Each horizontal segment gives the range of attainable gray levels in mode BR associated with each gray level in mode T.

To better illustrate this, let us consider the primary hypercolors. If we want to produce a gray level of 0.6 in mode T with this multiplexing palette, only a few metasurfaces are available. These are represented by the blue dots appearing on top of the upper gray line in Figure 7.6. Similarly, the nanostructures available to produce a 0.4 gray can be found on the lower gray line. By looking at the respective metasurfaces and their gray level in mode BR, one can confirm that the range of lightness is larger in the case of  $L_T = 0.6$  than in the case where  $L_T = 0.4$ .

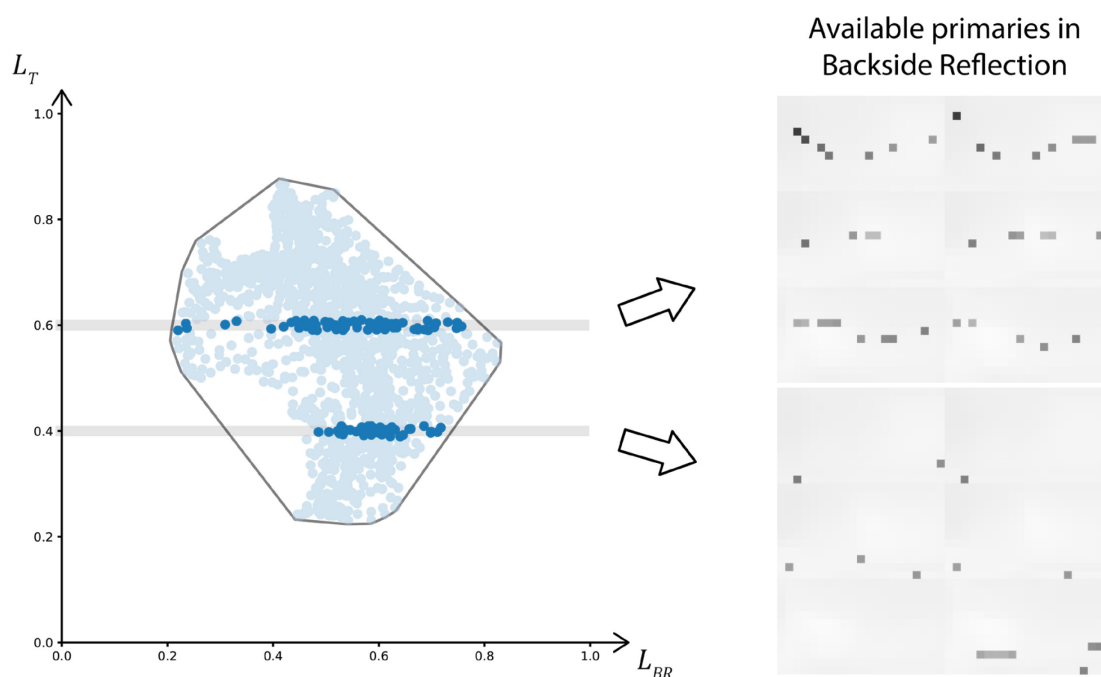


Figure 7.6 Two conditional gamuts shown on the 2D multiplexing plot. The images on the right side show the gray levels of the available primaries in mode “backside reflection” (BR) and their position in the palette where all other metasurfaces are greyed out.

This remains true with halftoning: for each gray level of mode T the range of attainable gray levels in mode BR is defined by the segment of the horizontal line that lies inside the convex hull. It is also symmetric: for each gray level in mode BR, the nanostructures generating this gray level lie on a vertical line, and the available range of lightness in mode T can be read by looking at the limits of the segment that lies inside the convex hull.

Since we want to prevent ghost images in the multiplexed print, the range of grays in one mode cannot depend on the range of grays in the other modes. The condition for multiplexing two grayscale images is therefore to find gray ranges  $S_T$  and  $S_{BR}$  for which

all the pairs of grays are attainable. If we denote by  $C$  the convex hull of the palette, this can be expressed as follows:

$$\forall (l_T, l_{BR}) \in S_T \times S_{BR}, \quad (l_T, l_{BR}) \in C \tag{7.1}$$

The cartesian product  $S_T \times S_{BR}$  defines a rectangle in the multi-modal plot. Any point inside this rectangle can be reproduced with the multiplexing palette if it lies inside its convex hull  $C$ . Gray ranges that can be multiplexed can be represented as rectangles enclosed in the convex hull. Conversely, any rectangle inside the convex hull defines a couple of achievable ranges of gray levels. The original grayscale images can be mapped onto these achievable ranges in order to be reproduced without any visible ghost.

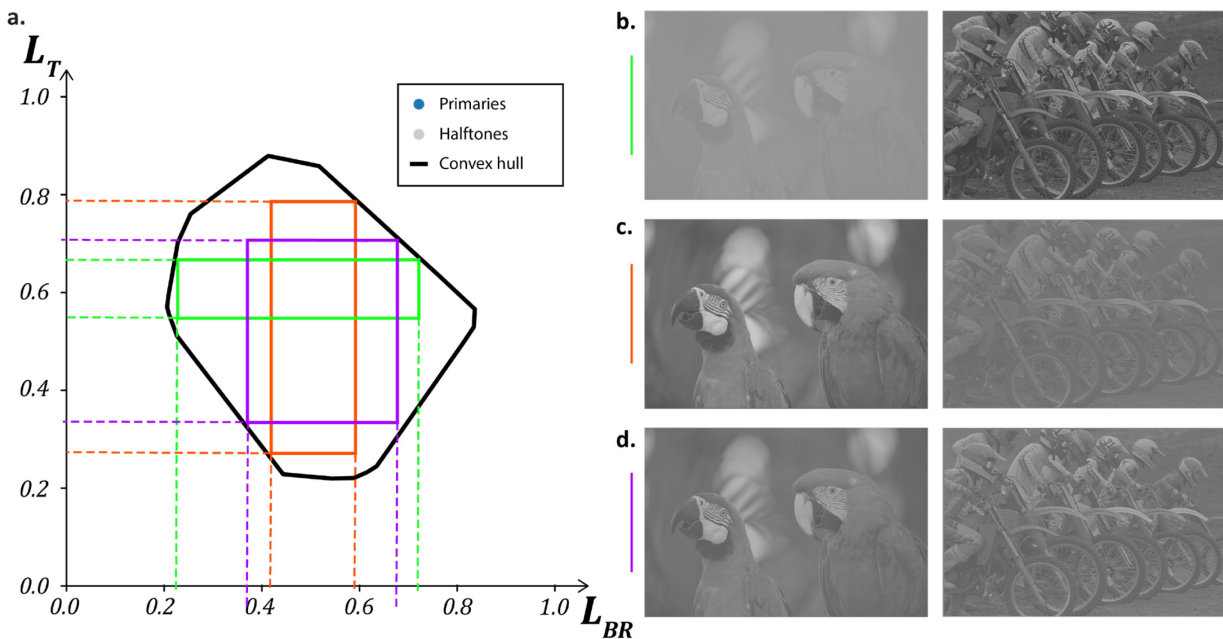


Figure 7.7 Several rectangles in the  $L_T \times L_{BR}$  plane (a) and the corresponding result of the mapping of two input images (b-d).

### 7.2.2 Optimum

Recognizing that the area of the rectangle is the geometrical mean of the maximum contrast attainable in each mode, one can see that maximizing the area of this rectangle allows maximizing the contrast of the produced images. The maximal contrast in both multiplexed modes is obtained when the ranges define the maximum area axis-aligned enclosed rectangle inside the convex hull of the multiplexing palette. Figure 7 shows several rectangles and the corresponding result of the mapping of two input images. Several algorithms exist to find this enclosed rectangle with maximal area [117]. Furthermore, if the contrast in one mode is a priority, it can be enhanced by introducing weights in the computation of the area.

### 7.2.3 Extension to M modes

This representation can be used to address more complex multiplexing cases: more modes, more colors. In the case of more modes, the generalization is direct: adding one mode adds one axis, and finding the rectangle becomes finding the rectangular cuboid (in the case of three modes) or hyperrectangle (N modes).

Each axis can not only represent grayscales, but also any gradients of color, for example from dark purple to beige. The only condition here is that color is expressed in a one-dimensional form. In this case the reasoning is the same, only the color conversion of the palette is different.

In the case of “full” color however, the number of dimensions is much higher, and the multiplexing condition has to be revised, as demonstrated in the following section.

## 7.3 Multicolor multi-mode gamut mapping

The case of grayscale images treated above was simple because the gray values in each mode could be represented along one axis. If we want to extend the representation to multi-color images, each axis must be converted into a full color 3D space. One may think it is equivalent to adding axes like what is done to generalize from 2 modes to M, but this is not true: the axes respect a hierarchical structure. With two modes, the representation remains 2D, but the axes are 3D, which is different from considering the multiplexing of 6 gray images where the representation would be 6D.

### 7.3.1 First case with reduced dimension

In order to show an example where a graphical representation is still possible, we propose to consider an example of multiplexing with two modes, in which the first image is a grayscale image (one axis), and the second is a two-dimensional color image (two axes). The conversion from a color image into a two-dimensional color image can be done in a variety of different ways. For this example, we took the two first channel of a RGB image and set the blue channel to 0.5. The resulting colors can be represented into the RG plane of the RGB color space. Figure 7.8 shows the colors contained in an image, and the ones contained in a palette of metasurfaces treated in this way. We could similarly generate 2-dimensional color images by fixing the gray value in the red channel or in the blue channel. Similar conversion is also possible in the CIE 1931 XYZ color space.

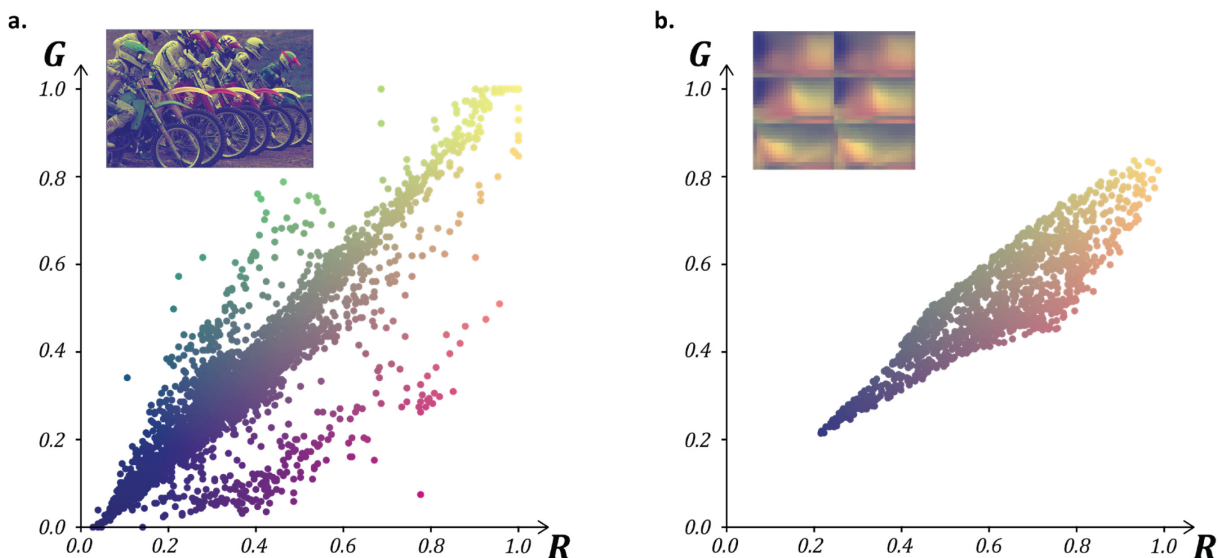


Figure 7.8 Color images projected to the  $(R, G, B=0.5)$  plane a. Test image from the Kodak PhotoCD Sampler. b. Palette image taken with our set-up.

When considering multiplexing with a grayscale image in the first mode T, the gray value chosen in this mode conditions the range of accessible colors in mode BR. Figure 7.9 shows sections of the palette in the  $(R, G, L_T)$  space. This three-dimensional space is still linear with respect to halftoning, and the attainable hypercolors can also be represented by the three-dimensional convex hull.

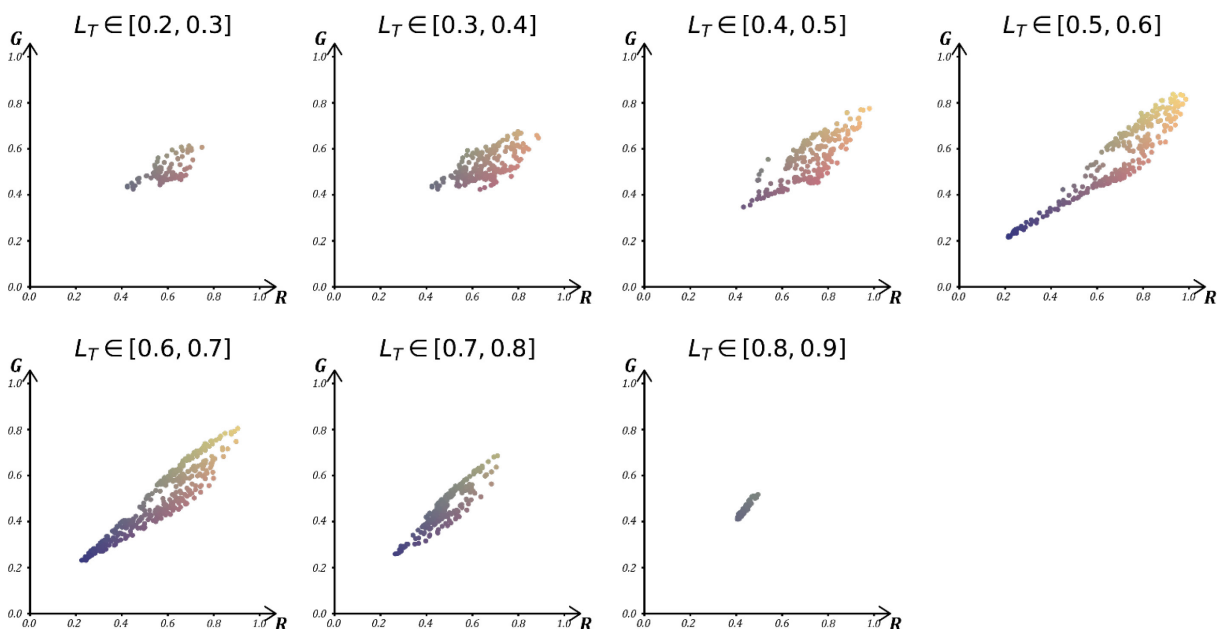
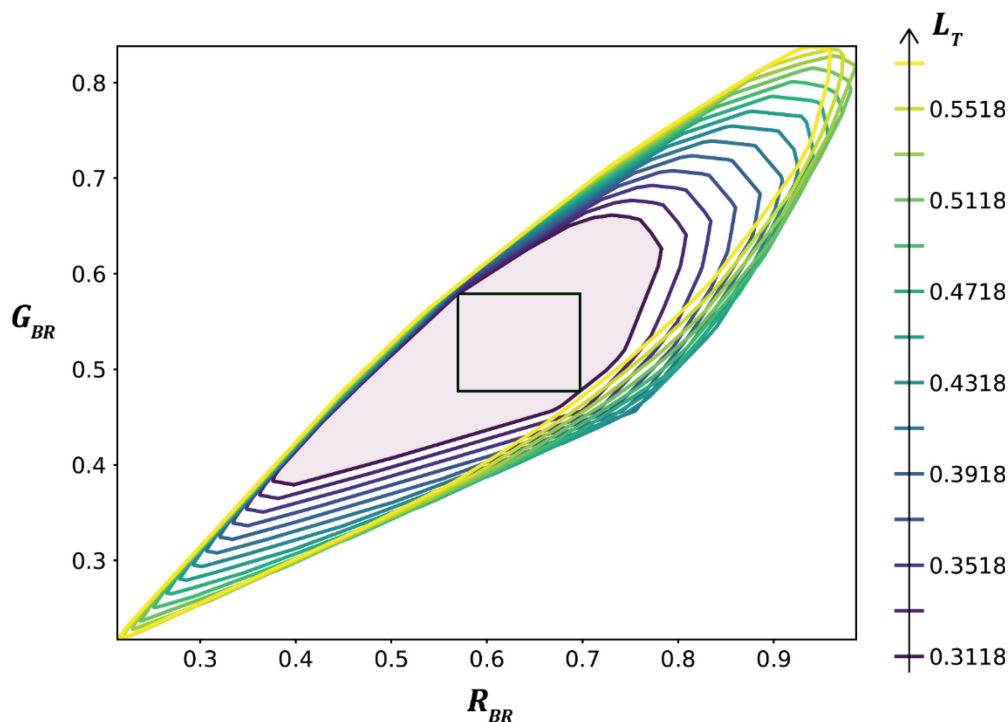


Figure 7.9 RG scatter plots of conditional gamuts for several grayscale values

Similarly to the grayscale example, intervals of  $R, G$  and  $L_T$  which allow multiplexing can be found and define a cuboid enclosed in the convex hull. But the cuboid is too

restrictive: it does not contain all the solutions for multiplexing. They are rather contained into a cylindroid whose base corresponds to the set of (R,G) value for which all  $L_T$  values in a certain range are accessible, i.e., in the convex hull.

Figure 7.10 shows the maximum enclosed box inside the convex hull, compared to the contour of the convex hull projected on the RG plane, for different  $L$  values inside the lightness interval covered by the box. All the inside area of this plot can be used for multiplexing with this  $L_T$  range, as it is included in the convex hull for all values of  $L_T$ .



*Figure 7.10. Contour plot representing a top-view of 3D convex hull associated with a multiplexing of a gray image and a 2-dimensional image. The top view of the maximum enclosed box is represented by the black rectangle. All points in this box are solutions for multiplexing. But the cylindroid whose base is the pink area, bigger than the box, also contains points that are solution for multiplexing.*

The same observation can be formulated in three dimensions in Figure 7.11, where both the convex hull and the enclosed box are projected onto three orthogonal planes to give an idea of their respective 3D shapes. Given a different section when projected on the RG plane (Figure 7.11c), the enclosed box could be extended further while remaining inside the convex hull.

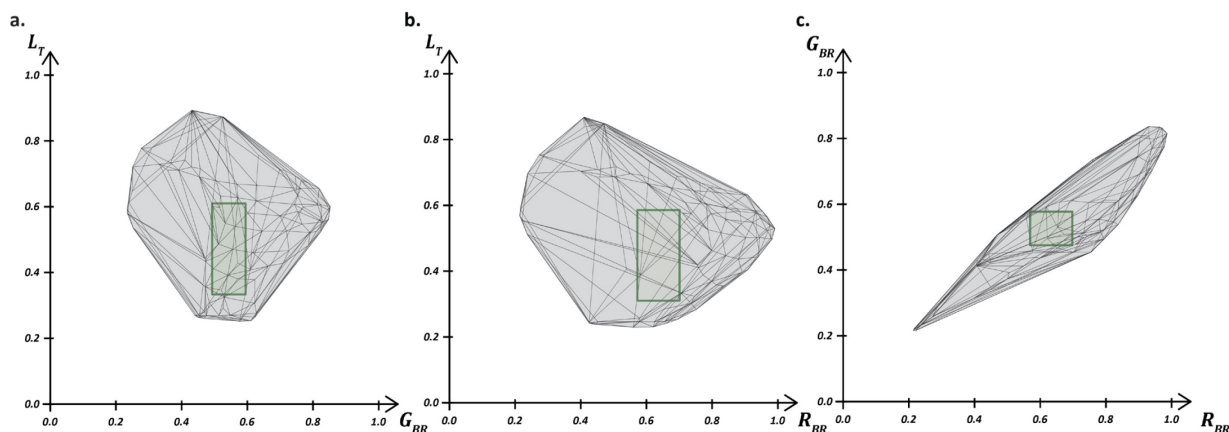


Figure 7.11 Maximum enclosed box inside the 3D convex hull (a) projected on the  $(L_T, G_{BR})$  plane (b) projected on the  $(L_T, R_{BR})$  plane (c) projected on the  $(R_{BR}, G_{BR})$  plane

We therefore know that we can find much more solutions for the multiplexing than the ones included inside the box, but finding them in an automated way, based on high-dimensional geometry tools, remains an open problem that we have not addressed yet. The multiplexed gamut mapping on these shapes might also not be straightforward. As the maximum enclosed box is still a solution for multiplexing, although not optimum in this case, we will seek to improve on this method in the following.

### 7.3.2 Change of basis and dimensionality reduction

In the case of  $M$  grayscale modes, the box has to be aligned with all the axes. As discussed above, in the RG plane, this axis alignment condition does not hold anymore. There are algorithms to find boxes that are not axis aligned, but their complexity increases dramatically with the number of dimensions. Besides, the box still has to be aligned with the  $L_T$  axis. A simpler solution is to perform a change of basis in each separate color space, to maximize the volume of the convex hull that can be covered by an axis-aligned box.

Moreover, a change of basis can be coupled with a projection in order to reduce the dimensionality of the data. Instead of converting some of the modes to grayscale, or selecting channels such as in the previous example, this projection can maximize the color variance that is retained in order to maximize the quality of the multiplexed images. Reducing the number of dimensions prior to the search of multiplexing solutions reduces the complexity of the problem and the difficulty of establishing the multiplexing conditions. It can therefore increase the reproduction quality of the original images.

Several algorithms exist for the change of basis and for dimensionality reduction. PCA is the most common one: it finds orthogonal axes to represent the data based on the eigenvectors of its covariance matrix[118]. In our example, when applying PCA to the RG colors of the palette, the axes found are:

$$PCA1_{BR} = -0.82R_{BR} - 0.58G_{BR} \quad (7.2)$$

$$PCA2_{BR} = 0.57R_{BR} - 0.82G_{BR} \quad (7.3)$$

Figure 7.12 shows the palette colors of mode BR in the  $(R_{BR}, G_{BR})$  plane (Figure 7.12a.) and in the  $(PCA1_{BR}, PCA2_{BR})$  plane after the change of basis (Figure 7.12b.). Performing the change of basis allows finding a larger box in the  $(L_T, PCA1_{BR}, PCA2_{BR})$  space when compared to the axis-aligned box that could have been found in the  $(L_T, R_{BR}, G_{BR})$  space. To show why, we project the box found in  $(L_T, PCA1_{BR}, PCA2_{BR})$  (Figure 7.12b.) onto the  $(R_{BR}, G_{BR})$  plane (Figure 7.12a.): the largest side of this rectangular section of the box is aligned with the maximum variance axis found by the PCA, allowing for a larger rectangular section than the box found in the  $(L_T, R_{BR}, G_{BR})$  space (Figure 7.11c).

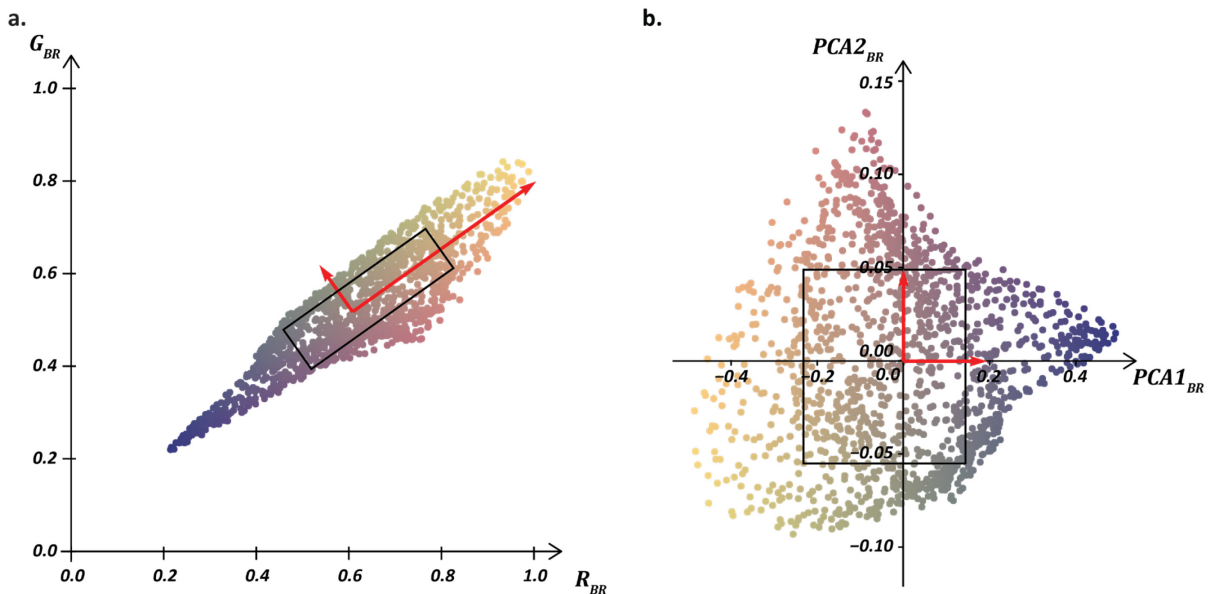


Figure 7.12 Palette colors (colored points) and maximum volume enclosed box found in the  $(L_T, PCA1_{BR}, PCA2_{BR})$  space, projected on (a) the  $(R_{BR}, G_{BR})$  plane (b) the  $(PCA1_{BR}, PCA2_{BR})$  plane (black rectangles).

*Red arrows show the PCA vectors scaled with respect to their respective explained variance in both planes. In each plot, their origin is set at the arithmetic mean of all the colors of the palette.*

The image to be multiplexed in this mode has to be transformed to the PCA basis before constructing the multiplexed image, by performing the gamut mapping in  $(L_T, PCA1_{BR}, PCA2_{BR})$ . The change of basis preserves the convexity and therefore mapping to the enclosed box in this space ensures the mapped image will be inside the convex hull even when transformed back to the  $(L_T, R_{BR}, G_{BR})$ .

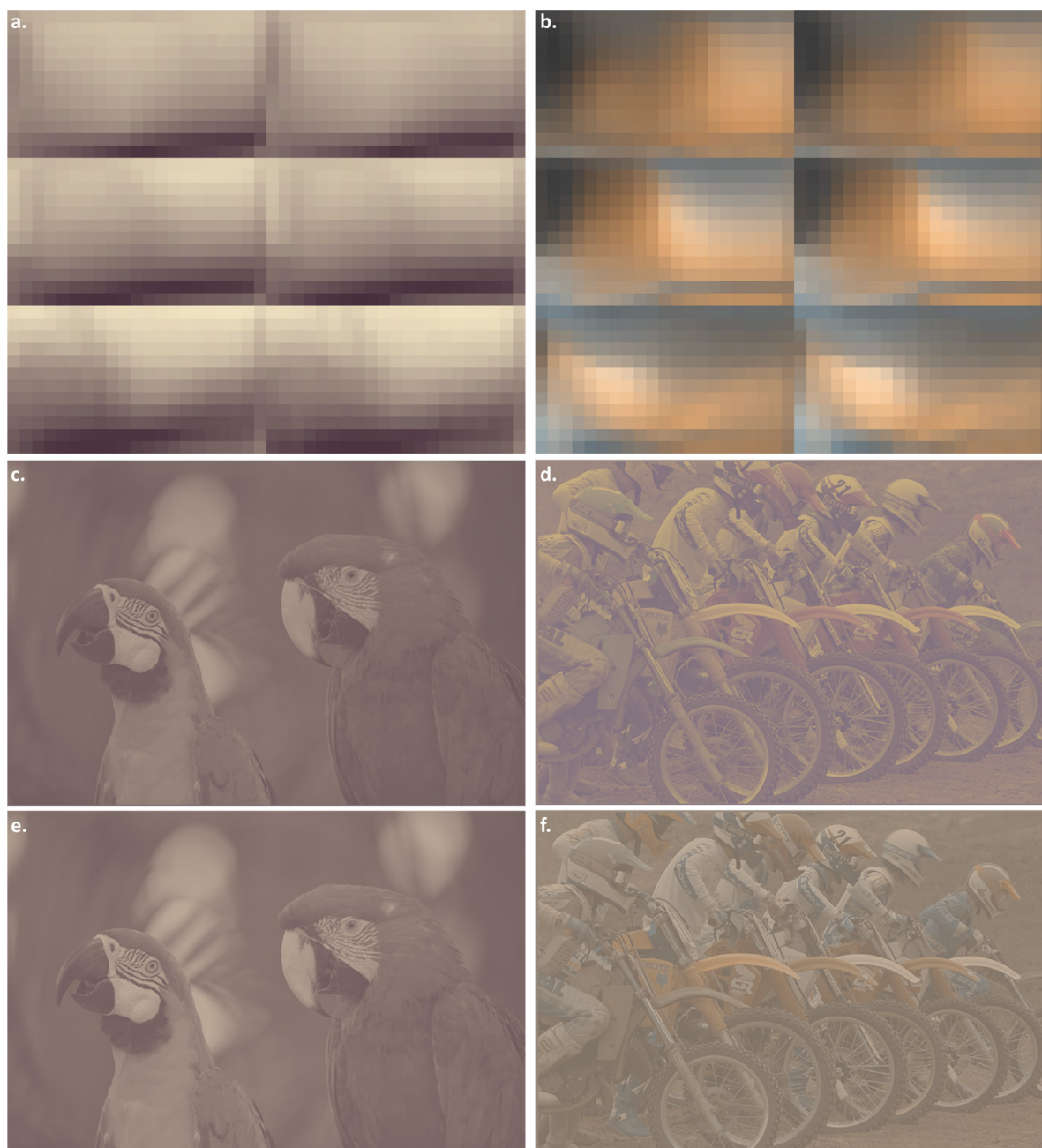
In this example, the change of basis allows a dramatic increase in the contrast of the BR image, while the T image retains the same contrast.



Figure 7.13 Images after multiplexed gamut mapping (a, b) without PCA (c, d) with PCA applied to the BR colors in the RG plane (b, d) transformed back to RGB for display.

In order to benefit from its dimensionality reduction as well, we can apply the PCA directly on the original palette RGB images. It must be trained independently in each of the modes, to preserve the multiplexing constraints between them. When applied to the full color space, up to three eigenvectors are found. Dimensionality reduction is obtained by ranking them by their explained variance (the ratio of the variance obtained when reconstructing the data from this vector only compared to the variance of the original data) and then by choosing the first vector or the two first ones. Although not discussed here, the same process can also be applied on higher-dimensional modes such as the result of multi- or hyper-spectral color measurement. In our 3D example, this can improve our multiplexing in two ways: it can be applied on the full-color space in mode T instead of a gray range, and/or it can be applied on mode BR directly from the RGB space, removing the first step of projecting on the RG plane. Figure 7.14a. shows the transformed palette in mode T after projecting on the first PCA axis  $PCA0 = -0.60R_T - 0.64G_T - 0.48B_T$ . Figure 7.14b. shows the transformed palette in mode BR after projecting on the two PCA axes  $PCA1' = 0.76R_{BR} + 0.55G_{BR} + 0.35B_{BR}$  and  $PCA2' = -0.53R_{BR} + 0.21G_{BR} + 0.81B_{BR}$ .





*Figure 7.14 Resulting images when PCA is used as for dimensionality reduction (a) T palette projected onto the PCA0 axis (b) BR palette projected onto the  $(PCA1', PCA2')$  plane (c, d) T and BR images scaled to the box found in the  $(PCA0, PCA1, PCA2)$  plane (e, f) T and BR images scaled to the box found in the  $(PCA0, PCA1', PCA2')$  plane*

As seen previously, perception of the final multiplexing result depends on the available palette and on the chosen images. However, we can notice a better performance in terms of volume of the convex hull covered by the multiplexed box with each improvement of the method (Table 7.1). Finding the axes that maximize the variance using PCA enables us to work in a space where the maximum enclosed box is closer to the optimum convex shape allowing multiplexing. Furthermore, the PCA's dimensionality

reduction abilities will be crucial when working in higher dimensions, as we will see in the next section.

*Table 7.1 Change of basis and dimensionality reduction strategies and corresponding maximum axis-aligned enclosed box volumes in the transformed space*

<b>PCA training space</b>	<b>PCA transformed Space</b>	<b>PCA used for:</b>	<b>Convex hull volume</b>	<b>Enclosed box volume (% of the convex hull)</b>
(L <sub>T</sub> , R <sub>BR</sub> , G <sub>BR</sub> )	-	-	0.0435	0.00359 (8.2%)
(L <sub>T</sub> , R <sub>BR</sub> , G <sub>BR</sub> )	(L <sub>T</sub> , PCA1 <sub>BR</sub> , PCA2 <sub>BR</sub> )	Change of basis	0.0435	0.00981 (22.5%)
(R <sub>T</sub> , G <sub>T</sub> , B <sub>T</sub> , R <sub>BR</sub> , G <sub>BR</sub> )	(PCA0 <sub>T</sub> , PCA1 <sub>BR</sub> , PCA2 <sub>BR</sub> )	Change of basis + Dimensionality reduction	0.0761	0.01739 (22.8%)
(R <sub>T</sub> , G <sub>T</sub> , B <sub>T</sub> , R <sub>BR</sub> , G <sub>BR</sub> , B <sub>BR</sub> )	(PCA0 <sub>T</sub> , PCA1' <sub>BR</sub> , PCA2' <sub>BR</sub> )	Change of basis + Dimensionality reduction	0.1549	0.04245 (27.3%)

### 7.3.3 Generalization to $N$ dimensions

For a better color rendering, it may seem interesting to keep more dimensions to preserve the color variations of the original images. However, in the case of multiplexing, the “curse of dimensionality” is that with higher dimensions comes a dramatic increase in the difficulty of the problem, which in turn affects the maximum contrast of the resulting multiplexed images. The non-optimality of the rectangular section of the box inside the color space of each mode, that we showed in the three-dimensional case, is also increased. Besides, a higher number of dimensions increases the algorithmic complexity of the search for the optimal box. Therefore, a trade-off is to be found between the final color variance and the complexity of the problem, by limiting the number of dimensions considered.

PCA can still be applied, to each mode independently, to reduce the complexity of the problem. The change of basis it provides is always beneficial even if a high number of dimensions is kept in the end.

## A.1 Continuous multiplexing workflow for full color multiplexing

The multi-mode gamut mapping yields better results when performed in a modified space by using PCA as a change of basis or dimensionality reduction in each mode. To be able to reproduce any color image in a multiplexed print with this method, a complete continuous multiplexing workflow, from the input images to the selection of the metasurfaces, has to be defined.

### 7.3.4 Application of dimensionality reduction without requiring the use of a device for observation

When reducing the dimensionality of the problem, the colors of the original images can be heavily transformed. However, the convexity induced by halftoning in a linear color space ensures that the colors obtained after gamut mapping in the dimensionality-reduced space can also be obtained by halftoning the real hypercolors. To guarantee that the final result will be observable without the use of an optical device, the images have to be transformed back to the respective color spaces of each mode before performing the halftoning.

The final continuous multiplexing workflow is thus as follows:

1. One mode is selected for each image, and the colors retrieved from the metasurface database are used to train PCA in each mode.
2. A number of dimensions is selected for each mode, resulting in a transformation of the database colors to the space created by the PCA. A multi-mode palette is created by concatenating the transformed colors in each mode.
3. The maximum volume box enclosed in the convex hull of the palette is computed in this transformed space.
4. The images are mapped to the maximum volume enclosed box in the transformed space.
5. The images and the palettes are transformed back to their original color spaces
6. The multi-mode image is halftoned with the multi-mode palette constructed in the original color spaces
7. The laser parameters are retrieved for each hypercolor selected by the multi-modal halftoning.
8. Laser processing

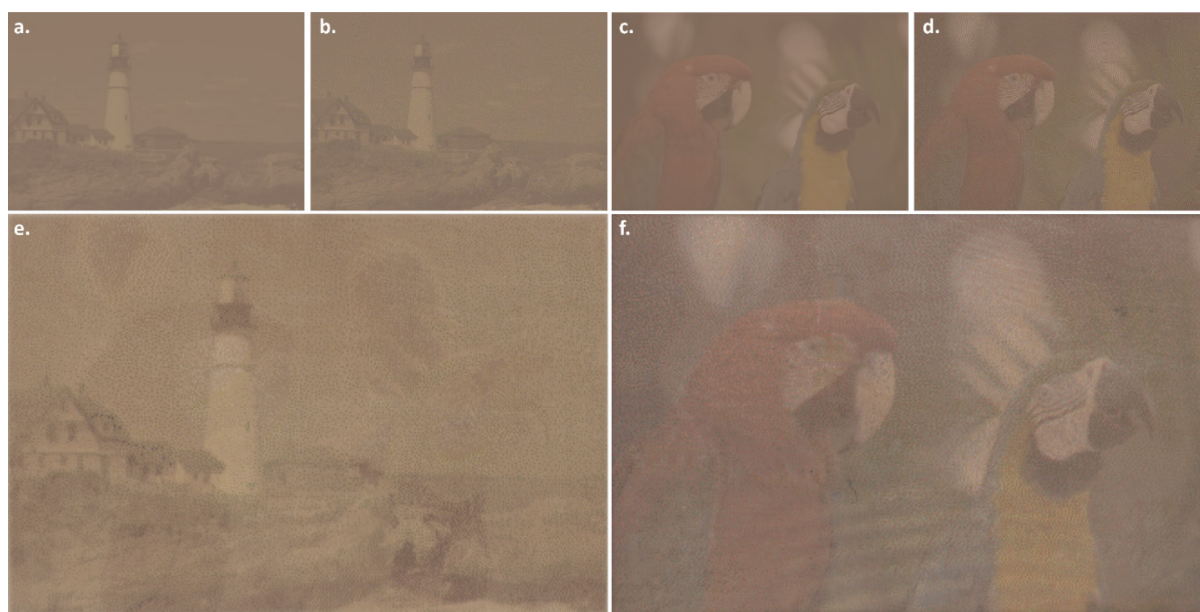
### 7.3.5 Number of primaries

In our workflow, multi-modal vector error diffusion can result in a high number of primaries. Depending on the processing technique used to produce the multiplexed print, this can be a problem. For example, with laser processing, a high number of primaries can induce a higher processing time due to switching delays between each set of laser parameters required to print each primary. The final number of primaries can be reduced, such as with regular color images, however with the risk of recreating ghost images. For this, we select a number of primaries such that the volume of the convex hull of the remaining palette in the multiplexed space is minimally reduced. An approximation

of this reduced palette can be obtained by performing a first multimodal halftoning of the considered images and removing the primaries that are little used in the resulting image.

#### 7.4 Experimental result

A first demonstrator of continuous multiplexing was produced using  $80^2 \mu\text{m}^2$  square pixels, to be observed in modes “transmission” and “backside reflection”. The original color images were adapted to our metasurface database using the dimensionality reduction workflow discussed above. PCA was used to reduce the dimensionality to 1 in mode “transmission” and as only a change of basis in mode “backside reflection”, keeping the full dimensionality of colors in this mode. Multi-mode gamut mapping was performed in the resulting four-dimensional space. The gamut-mapped images are shown in Figure 7.15a and c. Then, multi-mode halftoning was performed using the full palette of  $80^2 \mu\text{m}^2$  square pixels metasurfaces, resulting in images Figure 7.15b and d, which are produced by a total of 200 different metasurfaces.

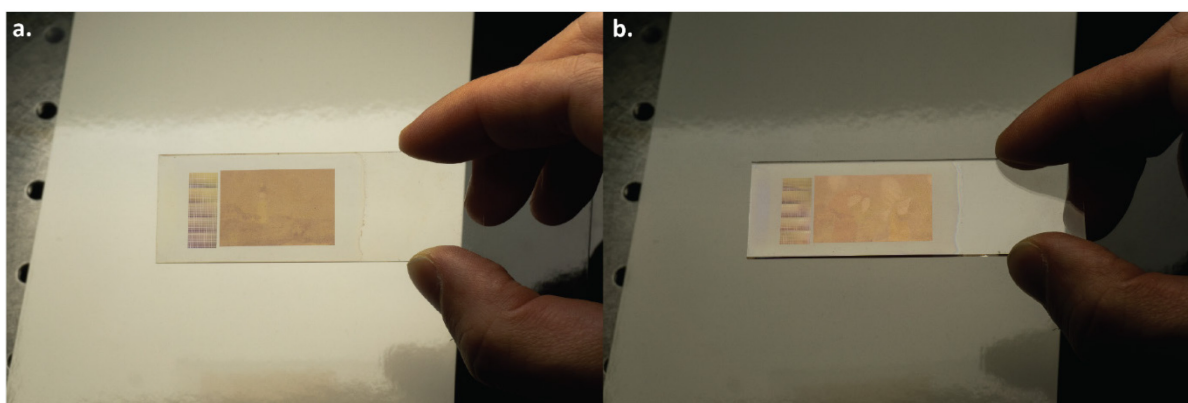


*Figure 7.15 First demonstrator of continuous multiplexing displaying a grayscale image in mode “transmission” and a color image in mode “backside reflection” (a, c) Images after multi-mode gamut mapping. (b, d) Multi-mode halftoning result with the color of the metasurfaces of our database and a 331x221 pixel resolution. (e, f) Color images of the demonstrator taken with our optical setup in modes “transmission” (e) and “backside reflection” (f). Due to the size of the printed image (26.48 mm x 17.68 mm) several color images were taken and assembled to produce these high resolution images of the print.*

This allowed us to produce a multiplexed print showing a grayscale image in mode “transmission” and a full color image in “backside reflection” with our technology. Figure 7.15e,f shows color photographs of the sample taken with our optical setup in both

modes. The large number of metasurfaces and the large size of the print resulted in a longer processing time of almost 20 minutes, which could be reduced by reducing the number of primaries in the multiplexed image (as mentioned in section 7.3.5).

Figure 7.16 shows the observation of the print from a distance with a commercial camera, allowing to perceive the actual size and viewing conditions of the image. The white balance and exposition settings are different in the two images to account for the eye adaptation when looking at either the transmitted image or the reflected image. [Supplementary Movie 6](#)[119] shows the dynamic observation of this demonstrator with a smartphone camera.



*Figure 7.16 Color photographs of the demonstrator observed from a distance showing its observation in real-life conditions. The sample is observed over a white paper. A LED table is placed at an angle above the viewing field. (a) When the sample is held perpendicular to the camera, the image of mode “transmission” is displayed thanks to the light reflected by the white paper and transmitted through the sample. (b) When tilting the sample, the light coming from the LED table is reflected by the print and the image of mode “backside reflection” is displayed. The size of the glass slide is 7 cm x 2 cm.*

## 7.5 Conclusion

The possibility of extending vector error diffusion to higher dimensional space opens the way for a color multiplexing workflow able to leverage the full potential of halftoning. In this chapter, we demonstrated the way vector error diffusion can be generalized to account for color reproduction in multiple modes. We use this new tool to derive a geometric representation of the design space available for multiplexing, using a multi-mode color space. This new representation enables to consider multiplexing in a continuous way, where the clustering step is not needed anymore. The range of colors available for multiplexing is a convex shape that can be found in this new representation. We show this convex shape in two low dimensionality spaces constructed with two modes which number of dimensions have been reduced to enable a graphical approach.

Using the algorithmic tools present in the literature, we propose a way of estimating the optimal range of colors available in each mode in the general case, by looking for a rectangular box in the multi-mode space. Observing the limitations of this approach in higher dimensional spaces, we propose a way of reducing the dimensionality of the problem using PCA. This also provides a possible change of basis maximizing the accordance between this rectangular box and the more general convex shape defined before, even when the dimensionality is preserved. This allows to propose a complete framework for continuous color multiplexing, by performing gamut mapping on the rectangular box and halftoning using multi-mode vector error diffusion. The performance of the different variations of this workflow are evaluated by considering the volume of the resulting multi-mode gamut.

Finally, this workflow is applied on the database created with our metasurfaces. We produced a demonstrator of a multiplexed print showing one image in full color, in mode “backside reflection” and a grayscale image in mode “transmission”. This first result of a color image multiplexing paves the way for more implementations where color images can be multiplexed in other modes, to be observed without the use of any additional device.

## Chapter 8.

# Conclusion and outlook

Image multiplexing is an interesting way to convey information on a physical medium while keeping this information readable by a human. Since its first appearances in holography and spatial-carrier photography in the 60s, its application towards security and identity documents has emerged as a highly interesting outcome. Printed image multiplexing, where the different images are displayed directly on the surface when the conditions of illumination and observation are varied is particularly adapted to these applications.

Although several works had already demonstrated printed image multiplexing, no general method to design a multiplexed print had been proposed until now. This is therefore what we set out to do, from the characterization of the printing process to the production of multiplexed images, while applying it to Ag:TiO<sub>2</sub> metasurfaces produced with a nanosecond laser. As those shown with femtosecond laser irradiation in previous works of the team, these metasurfaces were shown to display bright colors, reflection-transmission dichroism, polarization-dependent colors and diffract light thanks to gratings formed by self-organization of the Ag nanoparticles. This makes them very interesting for multiplexing.

To characterize the colors of these metasurfaces in these different modes, thousands of these were produced by varying the laser processing parameters in small increments and their color in each mode was measured using a dedicated color measurement setup. For a given metasurface, the set of colors produced in each mode defines a multi-dimensional color, which we named a *hypercolor*. We thus created a hypercolor database linking the colors of our metasurfaces to laser processing parameters used to produce them. This database allowed the production of full color images by adapting the color reproduction workflow of ink-based printing to be used with our metasurfaces. Any color image can be adapted to be printed with our technology through gamut mapping and vector error diffusion with the color of our metasurfaces.

To produce printed image multiplexing, however, the colors that can be selected for each image are strongly constrained. To find the colors that allow the demultiplexed images to be completely independent, we studied three different approaches, that we applied on our metasurface database. The two first methods are based on a discrete

approach, where the metasurfaces that display the same color in one of the modes are grouped in a first preprocessing step (clustering), while the last one removes this step by relying on high dimensional geometry algorithms to estimate the gamut available in each mode in a more continuous manner.

The first approach, described in Chapter 5, consisted in exploring the modes sequentially, considering in each new mode the colors that can be selected depending on the colors already selected in the previous modes. This allowed us to find solutions to multiplexing in 2 and 3 modes with two colors per mode by building logical color trees, and to produce multiple printed image multiplexing demonstrators, to be observable in a variety of modes. In particular, these demonstrators included the first-ever multiplexed print showing three images under white light where no additional observation device is needed.

The second approach, described in Chapter 6, relied on concepts from graph theory to represent hypercolor databases. In this graph representation, a specific kind of subgraphs called 2-partite cliques was found to indicate groups of colors that are solutions to 2-mode multiplexing. By searching for these 2-partite cliques using a recent algorithm, we were able to find 2-mode multiplexing solutions with a large number of colors in one of the modes. This allows to reproduce color variations in the multiplexed images, but the quality of the solutions is highly dependent on the preprocessing step of clustering, which would require further investigation.

Finally, the last approach relied on the generalization of gamut mapping and halftoning to multiplexing, by reasoning in a high dimensional space describing the relations between the colors in each mode in a more continuous way. By defining a rectangular box inside the multi-mode volume defined by the colors of our metasurface database, we were able to identify continuous ranges of colors where images can be multiplexed without generating any ghost image. An optimization algorithm, a change of basis and a possible dimensionality reduction provided by PCA, enabled to maximize the size of this box, thus maximizing the contrast and color fidelity of the multiplexed images. With this procedure, we were able to produce a 2-mode multiplexing demonstrator, where a full color image can be observed in mode “backside reflection” while a grayscale image is observed in mode “transmission”.

These three approaches, along with our color characterization procedure, can be applied to any technology producing mode-dependent colors. No model predicting the produced colors from the printing parameters is required. As long as a good sampling of the printing parameters can be produced experimentally, our characterization setup allows us to measure the color of a large number of samples rapidly in a variety of modes. If a prediction model was available, the hypercolor database could also be generated programmatically. However, the selection of the parameter sets to be included in the database is crucial. If the sampling of the parameters is not fine enough, or if some ranges



of parameters have been excluded, some of the possible hypercolors which might be needed for multiplexing could be missing from the database, which in turn could prevent from finding multiplexing solutions. Consequently, a perspective of this work could be to inverse the approaches described in this work to suggest new parameters to include in the database to expand its multiplexing capabilities.

Further, if the printing material is changed, for example in our case, if the initial state of the Ag:TiO<sub>2</sub> were to be modified, the solutions found on our metasurface database would not be valid anymore, and a new complete database needs to be inscribed, characterized, and searched for multiplexing solutions. One question that arises is: Can we adapt the found solutions to a new material, by characterizing only a few new samples made with the same printing process on this new material? A promising approach for this issue could be found in machine learning. Indeed, some preliminary work in this direction was carried out by the team by training a neural network to predict the spectra of our metasurfaces in each mode. This network gives spectral predictions that are within the acceptable perceptual distances in mode “backside reflection”, “transmission”, and “polarized transmission”. A process called *transfer learning*[120], where only part of the neural network is trained on a new database, thus requiring fewer samples to reach the same precision on a new material, could help adapt the results and extrapolate a new database from a few metasurfaces inscribed and characterized on the new material.

Finally, as mentioned in Chapter 3, the continuous improvement of color appearance models, and their adaptation to observation conditions, and materials, out of the ordinary matte reflective surfaces observed under diffuse illumination could help improve the perceptual precision of our different multiplexing solutions. Additionally, moving towards image color prediction models, such as iCAM[121] can help predict the appearance of ghost images. Indeed, it was for example observed that images containing large homogeneous areas, such as a piece of blue sky, were more prone to ghosting effects in this area. Further, the experimental variability of the colors, although possibly reduced to a minimum by further improvements of the sample preparation process, can be taken into account in these models, and thus lead to improvements in the design of the images.

There are still some developments ahead to bring this printed image technology to its full potential, and to a real world application in identity and secure documents. With this work, we hope to have opened the way to further improvements, and to its application to different technologies, some of which might provide even more modes or brighter colors. Our nanosecond laser processing of Ag:TiO<sub>2</sub> itself prove to be promising for this application, and deeper investigation into the physical processes that make this printed image multiplexing possible will be continued.

# References

- [1] C. J. de Larivière, “Du sceau au passeport: genèse des pratiques médiévales de l'identification,” p. 17, 2019.
- [2] R. L. van Renesse, J. van den Berg, and R. Tadema Wielandt, “Detection and integration of security devices in documents,” Los Angeles, CA, Apr. 1990, pp. 102–111. doi: 10/bjzh7c.
- [3] A. Ramos, W. Scott, W. Scott, D. Lloyd, K. O’Leary, and J. Waldo, “A threat analysis of RFID passports,” *Commun. ACM*, vol. 52, no. 12, pp. 38–42, Dec. 2009, doi: 10/dw36cd.
- [4] V. J. Cadarso, S. Chosson, K. Sidler, R. D. Hersch, and J. Brugger, “High-resolution 1D moirés as counterfeit security features,” *Light Sci. Appl.*, vol. 2, no. 7, pp. e86–e86, Jul. 2013, doi: 10.1038/lssa.2013.42.
- [5] V. Ostromoukhov, N. Rudaz, I. Amidror, P. Emmel, and R. D. Hersch, “Anticounterfeiting features of artistic screening,” Berlin, Germany, Dec. 1996, pp. 126–133. doi: 10/ctmwbn.
- [6] G. Pfaff, “Special effect pigments,” *Phys. Sci. Rev.*, May 2021, doi: 10/gmrtx7.
- [7] R. W. Phillip and A. F. Bleikolm, “Optical coatings for document security,” *Appl. Opt.*, vol. 35, no. 28, p. 5529, Oct. 1996, doi: 10.1364/AO.35.005529.
- [8] R. L. van Renesse, “The Use of Laser Technology to Protect Passport Photographs,” *Keesing J. Doc.*, no. 2, pp. 12–15, 2003.
- [9] D. Nébouy, M. Hébert, T. Fournel, N. Larina, and J.-L. Lesur, “Prediction of the spectral reflectance of laser-generated color prints by combination of an optical model and learning methods,” *J. Opt. Soc. Am. A*, vol. 32, no. 9, p. 1661, Sep. 2015, doi: 10.1364/JOSAA.32.001661.
- [10] M. Li *et al.*, “Optical modeling of changeable laser image functionality with analysis of the viewing performance,” *Appl. Opt.*, vol. 54, no. 20, p. 6162, Jul. 2015, doi: 10/gm4z43.
- [11] R. L. van Renesse, “Synergistic combination of document security techniques,” San Jose, CA, Apr. 2000, pp. 126–138. doi: 10/b7d9pp.
- [12] S. Bayda, M. Adeel, T. Tuccinardi, M. Cordani, and F. Rizzolio, “The History of Nanoscience and Nanotechnology: From Chemical-Physical Applications to Nanomedicine,” *Molecules*, vol. 25, no. 1, p. 112, Dec. 2019, doi: 10/ghdhdw.
- [13] Z. Liu *et al.*, “Three-Dimensional Self-Organization in Nanocomposite Layered Systems by Ultrafast Laser Pulses,” *ACS Nano*, vol. 11, no. 5, pp. 5031–5040, May 2017, doi: 10.1021/acsnano.7b01748.

- 
- [14] E. Rochat, S. D. Walker, and M. C. Parker, "Polarisation and wavelength division multiplexing at 1.55 $\mu$ m for bandwidth enhancement of multimode fibre based access networks," p. 13, 2004.
- [15] R. Franzen, "True Color Kodak Images," May 2021. <http://r0k.us/graphics/kodak/> (accessed Feb. 22, 2021).
- [16] "Image Encoding in Modulated Gratings from 1899 to 1970," *Opt. Acta Int. J. Opt.*, vol. 17, no. 8, pp. 631–635, Aug. 1970, doi: 10/cqw4zv.
- [17] C. P. Grover, "New Technique of Image Multiplexing Using Random Diffuser," *J. Opt. Soc. Am.*, vol. 62, no. 9, p. 1071, Sep. 1972, doi: 10/fs8wv4.
- [18] M. Papas, T. Houit, D. Nowrouzezahrai, M. Gross, and W. Jarosz, "The magic lens: refractive steganography," *ACM Trans. Graph.*, vol. 31, no. 6, p. 1, Nov. 2012, doi: 10.1145/2366145.2366205.
- [19] K. Sakurai, Y. Dobashi, K. Iwasaki, and T. Nishita, "Fabricating reflectors for displaying multiple images," *ACM Trans. Graph.*, vol. 37, no. 4, pp. 1–10, Jul. 2018, doi: 10.1145/3197517.3201400.
- [20] E. N. Leith and J. Upatnieks, "Wavefront Reconstruction with Diffused Illumination and Three-Dimensional Objects\*," *J. Opt. Soc. Am.*, vol. 54, no. 11, p. 1295, Nov. 1964, doi: 10/ftvdtq.
- [21] E. N. Leith, A. Kozma, J. Upatnieks, J. Marks, and N. Massey, "Holographic Data Storage in Three-Dimensional Media," *Appl. Opt.*, vol. 5, no. 8, p. 1303, Aug. 1966, doi: 10/bb2cht.
- [22] "Holographic storage isn't dead yet | Chips | Geek.com," Sep. 28, 2013. <https://web.archive.org/web/20130928132803/http://www.geek.com/chips/holographic-storage-isnt-dead-yet-547877/> (accessed Oct. 20, 2021).
- [23] H. J. Caulfield, "Wavefront Multiplexing by Holography," *Appl. Opt.*, vol. 9, no. 5, p. 1218, May 1970, doi: 10/dnhz5r.
- [24] S. C. Som and R. A. Lessard, "New Technique for Holographic Multiplexing\*†," *JOSA*, vol. 61, no. 9, pp. 1240–1245, Sep. 1971, doi: 10/cr2kcg.
- [25] M. J. Verheijen, "E-beam Lithography for Digital Holograms," *J. Mod. Opt.*, vol. 40, no. 4, pp. 711–721, Apr. 1993, doi: 10/cgs6tc.
- [26] Y.-W. Huang *et al.*, "Aluminum Plasmonic Multicolor Meta-Hologram," *Nano Lett.*, vol. 15, no. 5, pp. 3122–3127, May 2015, doi: 10.1021/acs.nanolett.5b00184.
- [27] W. Wan, J. Gao, and X. Yang, "Full-Color Plasmonic Metasurface Holograms," *ACS Nano*, vol. 10, no. 12, pp. 10671–10680, Dec. 2016, doi: 10.1021/acsnano.6b05453.
- [28] U. Levy, E. Marom, and D. Mendlovic, "Simultaneous multicolor image formation with a single diffractive optical element," *Opt. Lett.*, vol. 26, no. 15, pp. 1149–1151, 2001.
- [29] B. Wang *et al.*, "Visible-Frequency Dielectric Metasurfaces for Multiwavelength Achromatic and Highly Dispersive Holograms," *Nano Lett.*, vol. 16, no. 8, pp. 5235–5240, Aug. 2016, doi: 10.1021/acs.nanolett.6b02326.
- [30] W. Ye *et al.*, "Spin and wavelength multiplexed nonlinear metasurface holography," *Nat. Commun.*, vol. 7, no. 1, p. 11930, Sep. 2016, doi: 10.1038/ncomms11930.

- [31] W. T. Chen *et al.*, “High-Efficiency Broadband Meta-Hologram with Polarization-Controlled Dual Images,” *Nano Lett.*, vol. 14, no. 1, pp. 225–230, Jan. 2014, doi: 10.1021/nl403811d.
- [32] J. P. Balthasar Mueller, N. A. Rubin, R. C. Devlin, B. Groever, and F. Capasso, “Metasurface Polarization Optics: Independent Phase Control of Arbitrary Orthogonal States of Polarization,” *Phys. Rev. Lett.*, vol. 118, no. 11, p. 113901, Mar. 2017, doi: 10.1103/PhysRevLett.118.113901.
- [33] L. Jin *et al.*, “Noninterleaved Metasurface for ( $2^6-1$ ) Spin- and Wavelength-Encoded Holograms,” *Nano Lett.*, vol. 18, no. 12, pp. 8016–8024, Dec. 2018, doi: 10.1021/acs.nanolett.8b04246.
- [34] Y. Montelongo, J. O. Tenorio-Pearl, W. I. Milne, and T. D. Wilkinson, “Polarization Switchable Diffraction Based on Subwavelength Plasmonic Nanoantennas,” *Nano Lett.*, vol. 14, no. 1, pp. 294–298, Jan. 2014, doi: 10.1021/nl4039967.
- [35] B. Wang *et al.*, “Polarization-controlled color-tunable holograms with dielectric metasurfaces,” *Optica*, vol. 4, no. 11, p. 1368, Nov. 2017, doi: 10.1364/OPTICA.4.001368.
- [36] S. M. Kamali, E. Arbabi, A. Arbabi, Y. Horie, M. Faraji-Dana, and A. Faraon, “Angle-Multiplexed Metasurfaces: Encoding Independent Wavefronts in a Single Metasurface under Different Illumination Angles,” *Phys. Rev. X*, vol. 7, no. 4, p. 041056, Dec. 2017, doi: 10.1103/PhysRevX.7.041056.
- [37] J. Deng *et al.*, “Spatial Frequency Multiplexed Meta-Holography and Meta-Nanoprinting,” *ACS Nano*, vol. 13, no. 8, pp. 9237–9246, Aug. 2019, doi: 10.1021/acsnano.9b02611.
- [38] H. Ren, X. Li, Q. Zhang, and M. Gu, “On-chip noninterference angular momentum multiplexing of broadband light,” *Science*, vol. 352, no. 6287, pp. 805–809, May 2016, doi: 10.1126/science.aaf1112.
- [39] Y. Hu *et al.*, “3D-Integrated metasurfaces for full-colour holography,” *Light Sci. Appl.*, vol. 8, no. 1, Dec. 2019, doi: 10.1038/s41377-019-0198-y.
- [40] K. T. P. Lim, H. Liu, Y. Liu, and J. K. W. Yang, “Holographic colour prints for enhanced optical security by combined phase and amplitude control,” *Nat. Commun.*, vol. 10, no. 1, Dec. 2019, doi: 10.1038/s41467-018-07808-4.
- [41] L. Huang, S. Zhang, and T. Zentgraf, “Metasurface holography: from fundamentals to applications,” *Nanophotonics*, vol. 7, no. 6, pp. 1169–1190, Jun. 2018, doi: 10.1515/nanophotonics-2017-0112.
- [42] V. Babaei and R. D. Hersch, “Color reproduction of metallic-ink images,” *J. Imaging Sci. Technol.*, vol. 60, no. 3, pp. 30503–1, 2016.
- [43] R. Rossier and R. D. Hersch, “Hiding patterns with daylight fluorescent inks,” in *Color and Imaging Conference*, 2011, vol. 2011, pp. 223–228.
- [44] V. Žiljak, K. Pap, and I. Žiljak, “CMYKIR security graphics separation in the infrared area,” *Infrared Phys. Technol.*, vol. 52, no. 2–3, pp. 62–69, Mar. 2009, doi: 10.1016/j.infrared.2009.01.001.
- [45] P. Pjanic and R. D. Hersch, “Color imaging and pattern hiding on a metallic substrate,” *ACM Trans. Graph.*, vol. 34, no. 4, p. 130:1–130:10, Jul. 2015, doi: 10.1145/2766944.

- [46] N. Dalloz, S. Mazauric, T. Fournel, and M. Hébert, "How to design a recto-verso print displaying different images in various everyday-life lighting conditions," in *Electronic Imaging*, Jan. 2017, vol. 2017, pp. 33–41. doi: 10/gmkgm6.
- [47] P. Zijlstra, J. W. M. Chon, and M. Gu, "Five-dimensional optical recording mediated by surface plasmons in gold nanorods," *Nature*, vol. 459, no. 7245, pp. 410–413, May 2009, doi: 10.1038/nature08053.
- [48] Y. Bao *et al.*, "Coherent Pixel Design of Metasurfaces for Multidimensional Optical Control of Multiple Printing-Image Switching and Encoding," *Adv. Funct. Mater.*, vol. 28, no. 51, p. 1805306, 2018, doi: 10.1002/adfm.201805306.
- [49] M. Song, Z. A. Kudyshev, H. Yu, A. Boltasseva, V. M. Shalaev, and A. V. Kildishev, "Achieving full-color generation with polarization-tunable perfect light absorption," *Opt. Mater. Express*, vol. 9, no. 2, pp. 779–787, Feb. 2019, doi: 10.1364/OME.9.000779.
- [50] L. Wang, T. Li, R. Y. Guo, W. Xia, X. G. Xu, and S. N. Zhu, "Active display and encoding by integrated plasmonic polarizer on light-emitting-diode," *Sci. Rep.*, vol. 3, no. 1, Dec. 2013, doi: 10.1038/srep02603.
- [51] H. Liu *et al.*, "Tunable Resonator-Upconverted Emission (TRUE) Color Printing and Applications in Optical Security," *Adv. Mater.*, vol. 31, no. 15, p. 1807900, Apr. 2019, doi: 10.1002/adma.201807900.
- [52] X. M. Goh *et al.*, "Three-dimensional plasmonic stereoscopic prints in full colour," *Nat. Commun.*, vol. 5, p. 5361, Nov. 2014, doi: 10.1038/ncomms6361.
- [53] E. Heydari, J. R. Sperling, S. L. Neale, and A. W. Clark, "Plasmonic Color Filters as Dual-State Nanopixels for High-Density Microimage Encoding," *Adv. Funct. Mater.*, vol. 27, no. 35, p. 1701866, Sep. 2017, doi: 10.1002/adfm.201701866.
- [54] N. Destouches *et al.*, "Laser-Empowered Random Metasurfaces for White Light Printed Image Multiplexing," *Adv. Funct. Mater.*, vol. 31, no. 18, p. 2010430, Feb. 2021, doi: 10.1002/adfm.202010430.
- [55] Q. Dai, N. Zhou, L. Deng, J. Deng, Z. Li, and G. Zheng, "Dual-Channel Binary Gray-Image Display Enabled with Malus-Assisted Metasurfaces," *Phys. Rev. Appl.*, vol. 14, no. 3, p. 034002, Sep. 2020, doi: 10.1103/PhysRevApplied.14.034002.
- [56] P. Pjanic and R. D. Hersch, "Color changing effects with anisotropic halftone prints on metal," *ACM Trans. Graph. TOG*, vol. 34, no. 6, p. 167, 2015.
- [57] N. Dalloz, S. Mazauric, T. Fournel, and M. Hébert, "How to design a recto-verso print displaying different images in various everyday-life lighting conditions," *Electron. Imaging*, vol. 2017, no. 8, pp. 33–41, Jan. 2017, doi: 10.2352/ISSN.2470-1173.2017.8.MAAP-289.
- [58] P. Pjanic, M. Shahpaski, and A. Grundhöfer, "Magic Prints: Image-Changing Prints Observed under Visible and 365 nm UV Light," *J. Imaging Sci. Technol.*, vol. 63, no. 2, pp. 20504-1-20504-14, Mar. 2019, doi: 10/gm3wh6.
- [59] S. J. Tan, X. M. Goh, Y. M. Wang, J. K. W. Yang, and J. Teng, "ENGINEERING PLASMONIC COLORS IN METAL NANOSTRUCTURES," *J. Mol. Eng. Mater.*, vol. 02, no. 02, p. 1440011, Jun. 2014, doi: 10/gm4z7g.

- [60] V. Veiko *et al.*, "Controlled oxide films formation by nanosecond laser pulses for color marking," *Opt. Express*, vol. 22, no. 20, p. 24342, Oct. 2014, doi: 10.1364/OE.22.024342.
- [61] Ya. M. Andreeva *et al.*, "Laser coloration of metals in visual art and design," *Opt. Mater. Express*, vol. 9, no. 3, p. 1310, Mar. 2019, doi: 10/gm42jc.
- [62] H. Roozbahani, M. Alizadeh, H. Handroos, and A. Salminen, "Color Laser Marking: Repeatability, Stability and Resistance Against Mechanical, Chemical and Environmental Effects," *IEEE Access*, vol. 8, pp. 214196–214208, 2020, doi: 10/gm5n7t.
- [63] H. Liu, W. Lin, and M. Hong, "Surface coloring by laser irradiation of solid substrates," *APL Photonics*, vol. 4, no. 5, p. 051101, May 2019, doi: 10/gjj5x3.
- [64] V. Veiko *et al.*, "Development of complete color palette based on spectrophotometric measurements of steel oxidation results for enhancement of color laser marking technology," *Mater. Des.*, vol. 89, pp. 684–688, Jan. 2016, doi: 10/f73dfr.
- [65] S. Cucerca, P. Didyk, H.-P. Seidel, and V. Babaei, "Computational Image Marking on Metals via Laser Induced Heating," *ACM Trans. Graph.*, vol. 39, no. 4, p. 12, Aug. 2020, doi: 10.1145/3386569.3392423.
- [66] J.-M. Guay *et al.*, "Laser-induced plasmonic colours on metals," *Nat. Commun.*, vol. 8, no. 1, p. 16095, Dec. 2017, doi: 10/gbptsm.
- [67] G. V. Odintsova *et al.*, "High-resolution large-scale plasmonic laser color printing for jewelry applications," *Opt. Express*, vol. 27, no. 3, p. 3672, Feb. 2019, doi: 10/gmq75s.
- [68] B. Dusser *et al.*, "Controlled nanostructures formation by ultra fast laser pulses for color marking," *Opt. Express*, vol. 18, no. 3, p. 2913, Feb. 2010, doi: 10.1364/OE.18.002913.
- [69] M. D. Ooms, Y. Jeyaram, and D. Sinton, "Disposable Plasmonics: Rapid and Inexpensive Large Area Patterning of Plasmonic Structures with CO<sub>2</sub> Laser Annealing," *Langmuir*, vol. 31, no. 18, pp. 5252–5258, May 2015, doi: 10/gmq8js.
- [70] A. S. Roberts *et al.*, "Laser writing of bright colours on near-percolation plasmonic reflector arrays," vol. 13, no. 1, pp. 71–77, Dec. 2018.
- [71] Y. Kuroiwa and T. Tatsuma, "Laser Printing of Translucent Plasmonic Full-Color Images with Transmission-Scattering Dichroism of Silver Nanoparticles," *ACS Appl. Nano Mater.*, vol. 3, no. 3, pp. 2472–2479, Mar. 2020, doi: 10/gm5wkk.
- [72] X. Zhu, W. Yan, U. Levy, N. A. Mortensen, and A. Kristensen, "Resonant laser printing of structural colors on high-index dielectric metasurfaces," *Sci. Adv.*, vol. 3, no. 5, p. e1602487, May 2017, doi: 10.1126/sciadv.1602487.
- [73] Y. Zhang *et al.*, "Full-visible multifunctional aluminium metasurfaces by *in situ* anisotropic thermoplasmonic laser printing," *Nanoscale Horiz.*, vol. 4, no. 3, pp. 601–609, Jan. 2019, doi: 10/gkzpnq.
- [74] N. Crespo-Monteiro *et al.*, "One-Step Microstructuring of TiO<sub>2</sub> and Ag-TiO<sub>2</sub> Films by Continuous Wave Laser Processing in the UV and Visible Ranges," *J. Phys. Chem. C*, vol. 116, no. 51, pp. 26857–26864, Dec. 2012, doi: 10.1021/jp3096264.
- [75] N. Sharma, N. Destouches, C. Florian, R. Serna, and J. Siegel, "Tailoring metal-dielectric nanocomposite materials with ultrashort laser pulses for dichroic color

- control,” *Nanoscale*, vol. 11, no. 40, pp. 18779–18789, Oct. 2019, doi: 10.1039/C9NR06763A.
- [76] N. Destouches *et al.*, “Dichroic colored luster of laser-induced silver nanoparticle gratings buried in dense inorganic films,” *J. Opt. Soc. Am. B*, vol. 31, no. 11, p. C1, Nov. 2014, doi: 10.1364/JOSAB.31.0000C1.
- [77] N. Sharma *et al.*, “Laser-driven plasmonic gratings for hiding multiple images,” *Mater. Horiz.*, vol. 6, no. 5, pp. 978–983, Feb. 2019, doi: 10.1039/C9MH00017H.
- [78] N. Dalloz *et al.*, “Anti-Counterfeiting White Light Printed Image Multiplexing by Fast Nanosecond Laser Processing,” *Adv. Mater.*, p. 2104054, Nov. 2021, doi: 10.1002/adma.202104054.
- [79] R. D. Hersch, P. Donzé, and S. Chosson, “Color images visible under UV light,” *ACM Trans. Graph.*, vol. 26, no. 3, p. 10.
- [80] J. Morović, *Color Gamut Mapping*. John Wiley & Sons, 2008.
- [81] S. M. Chosson and R. D. Hersch, “Color gamut reduction techniques for printing with custom inks,” San Jose, CA, Dec. 2001, pp. 110–120. doi: 10.1117/12.452980.
- [82] B. M. D. Fairchild, *Color Appearance Models*. John Wiley & Sons Inc. Accessed: Feb. 23, 2018. [Online]. Available: <https://www.dawsonera.com/abstract/9781118653111>
- [83] T. Suzuki, “Color digital halftoning taking colorimetric color reproduction into account,” *J. Electron. Imaging*, vol. 5, no. 1, p. 97, Jan. 1996, doi: 10.1117/12.227441.
- [84] G. Sharma, *Digital color imaging handbook*. Boca Raton, FL: CRC Press, 2003. Accessed: Aug. 20, 2021. [Online]. Available: <http://www.crcnetbase.com/isbn/9781420041484>
- [85] Z. Fan, “Boundary artifacts reduction in vector error diffusion,” Dec. 1998, p. 5. doi: 10.1117/12.334592.
- [86] J. Morovic, V. Cheung, P. Morovic, and P. Company, “Why We Don’t Know How Many Colors There Are,” in *CGIV 2012 Final Program and Proceedings*, May 2012, pp. 49–53.
- [87] J. Luo, K. E. Spaulding, and Q. Yu, “Novel color palettization scheme for preserving important colors,” Santa Clara, CA, Jan. 2003, p. 409. doi: 10/btj7zg.
- [88] N. J. Mitra and M. R. Gupta, “Two-stage color palettization for error diffusion,” San Jose, CA, Jun. 2002, pp. 207–217. doi: 10/bsn9qj.
- [89] C. A. Bouman, “Optimized universal color palette design for error diffusion,” *J. Electron. Imaging*, vol. 4, no. 2, p. 131, Apr. 1995, doi: 10/czxtj5.
- [90] N. Dalloz and M. Hébert, “Conditional Color Gamut for Color Management of Multiview Printed Images,” in *Computational Color Imaging*, 2019, pp. 63–76.
- [91] Nicolas Dalloz, *Supplementary Movie 1*, (Nov. 17, 2021). Accessed: Nov. 19, 2021. [Online]. Available: <https://www.youtube.com/watch?v=komiLMP8IFg>
- [92] Nicolas Dalloz, *Supplementary Movie 2*, (Nov. 17, 2021). Accessed: Nov. 19, 2021. [Online]. Available: [https://www.youtube.com/watch?v=YnROPHL2\\_Es](https://www.youtube.com/watch?v=YnROPHL2_Es)
- [93] Nicolas Dalloz, *Supplementary Movie 3*, (Nov. 17, 2021). Accessed: Nov. 19, 2021. [Online]. Available: <https://www.youtube.com/watch?v=ApCQj7cKYkw>

- [94] Nicolas Dalloz, *Supplementary Movie 4*, (Nov. 17, 2021). Accessed: Nov. 19, 2021. [Online]. Available: <https://www.youtube.com/watch?v=qUC403aJZR8>
- [95] Nicolas Dalloz, *Supplementary Movie 5*, (Nov. 17, 2021). Accessed: Nov. 19, 2021. [Online]. Available: <https://www.youtube.com/watch?v=GFbsZaba0wE>
- [96] C. Phillips, K. Wang, E. Baker, J. Bubier, E. Chesler, and M. Langston, "On Finding and Enumerating Maximal and Maximum  $k$ -Partite Cliques in  $k$ -Partite Graphs," *Algorithms*, vol. 12, no. 1, p. 23, Jan. 2019, doi: 10.3390/a12010023.
- [97] R. Thomas, "An Update on the Four-Color Theorem," *Notices of the AMS*, vol. 45, no. 7, p. 12, 1998.
- [98] T. B. Boffey, *Graph Theory in Operations Research*. Macmillan International Higher Education, 1982.
- [99] O. Lezoray and L. Grady, *Image Processing and Analysis with Graphs: Theory and Practice*. CRC Press, 2017.
- [100] T. H. Cormen, Ed., *Introduction to algorithms*, 3rd ed. Cambridge, Mass: MIT Press, 2009.
- [101] C. A. Phillips, "Multipartite Graph Algorithms for the Analysis of Heterogeneous Data," University of Tennessee. [Online]. Available: [https://trace.tennessee.edu/utk\\_graddiss/3600](https://trace.tennessee.edu/utk_graddiss/3600)
- [102] C. Berge, *Hypergraphs: combinatorics of finite sets*. Amsterdam ; New York: North Holland : Distributors for the U.S.A. and Canada, Elsevier Science Pub. Co, 1989.
- [103] A. Antelmi *et al.*, "SimpleHypergraphs.jl—Novel Software Framework for Modelling and Analysis of Hypergraphs," in *Algorithms and Models for the Web Graph*, Cham, 2019, pp. 115–129. doi: 10/gmpdj7.
- [104] M. Alishahi, "Colorful Subhypergraphs in Uniform Hypergraphs," *Electron. J. Comb.*, vol. 24, no. 1, p. P1.23, Feb. 2017, doi: 10/gnzgx3.
- [105] R. M. Karp, "Reducibility among Combinatorial Problems," in *Complexity of Computer Computations*, R. E. Miller, J. W. Thatcher, and J. D. Bohlinger, Eds. Boston, MA: Springer US, 1972, pp. 85–103. doi: 10.1007/978-1-4684-2001-2\_9.
- [106] "Clique problem," *Wikipedia*. Aug. 25, 2021. Accessed: Sep. 06, 2021. [Online]. Available: [https://en.wikipedia.org/w/index.php?title=Clique\\_problem&oldid=1040554376](https://en.wikipedia.org/w/index.php?title=Clique_problem&oldid=1040554376)
- [107] M. R. Garey and D. S. Johnson, *Computers and intractability: a guide to the theory of NP-completeness*. San Francisco: Freeman, 2009.
- [108] R. Peeters, "The maximum edge biclique problem is NP-complete," *Discrete Appl. Math.*, vol. 131, no. 3, pp. 651–654, Sep. 2003, doi: 10/ctrpq9.
- [109] C. Phillips, "The Quest for Dense Structure in Heterogeneous Data Exploration," presented at the CiSML Seminar, Mar. 27, 2015.
- [110] C. Bron and J. Kerbosch, "Algorithm 457: finding all cliques of an undirected graph," *Commun. ACM*, vol. 16, no. 9, pp. 575–577, Sep. 1973, doi: 10/c6p8tj.



- [111] E. Tomita, A. Tanaka, and H. Takahashi, "The worst-case time complexity for generating all maximal cliques and computational experiments," *Theor. Comput. Sci.*, vol. 363, no. 1, pp. 28–42, Oct. 2006, doi: 10/d3vs7s.
- [112] F. N. Abu-Khzam, N. E. Baldwin, M. A. Langston, and N. F. Samatova, "On the Relative Efficiency of Maximal Clique Enumeration Algorithms, with Application to High-Throughput Computational Biology," Apr. 2018, p. 10.
- [113] F. Cazals and C. Karande, "A note on the problem of reporting maximal cliques," *Theor. Comput. Sci.*, vol. 407, no. 1–3, pp. 564–568, Nov. 2008, doi: 10/fszmdh.
- [114] A. A. Hagberg, D. A. Schult, and P. J. Swart, "Exploring Network Structure, Dynamics, and Function using NetworkX," in *Proceedings of the 7th Python in Science Conference (SciPy)*, 2008, pp. 11–15.
- [115] J. Gerhardt, J. Y. Hardeberg, and G. U. College, "Spectral Colour Reproduction by Vector Error Diffusion," in *CGIV 2006 Final Program and Proceedings*, Jan. 2006, pp. 469–473.
- [116] J. Gerhardt, "Reproduction spectrale de la couleur: approches par modélisation et par diffusion vectorielle de l'erreur," phdthesis, Télécom ParisTech, 2007. Accessed: May 28, 2021. [Online]. Available: <https://pastel.archives-ouvertes.fr/pastel-00003354>
- [117] N. Amenta, "Bounded boxes, Hausdorff distance, and a new proof of an interesting Helly-type theorem," in *Proceedings of the tenth annual symposium on Computational geometry - SCG '94*, Stony Brook, New York, United States, 1994, pp. 340–347. doi: 10.1145/177424.178064.
- [118] M. E. Tipping and C. M. Bishop, "Mixtures of Probabilistic Principal Component Analysers," *Neural Comput.*, vol. 11, no. 2, pp. 443–482, Feb. 1999.
- [119] Nicolas Dalloz, *Supplementary Movie 6*, (Nov. 19, 2021). Accessed: Nov. 19, 2021. [Online]. Available: <https://www.youtube.com/watch?v=HvxC6n6lXfU>
- [120] S. So, T. Badloe, J. Noh, J. Bravo-Abad, and J. Rho, "Deep learning enabled inverse design in nanophotonics," *Nanophotonics*, vol. 9, no. 5, pp. 1041–1057, May 2020, doi: 10/ghswgn.
- [121] M. D. Fairchild and G. M. Johnson, "Meet iCAM: A next-generation color appearance model," in *Color and Imaging Conference*, 2002, vol. 2002, pp. 33–38.
- [122] E. C. Carter *et al.*, "CIE 015:2018 Colorimetry, 4th Edition," International Commission on Illumination (CIE), Oct. 2018. doi: 10.25039/TR.015.2018.
- [123] "IEC 61966-2-1:1999 - multimedia systems and equipment - colour measurement and management - part 2-1: colour management - default rgb colour space - srgb," International Electrotechnical Commission, 1999. Accessed: Mar. 24, 2021. [Online]. Available: <https://webstore.iec.ch/publication/6169>
- [124] B. Lindbloom, "RGB/XYZ Matrices." [http://brucelindbloom.com/index.html?Eqn\\_RGB\\_XYZ\\_Matrix.html](http://brucelindbloom.com/index.html?Eqn_RGB_XYZ_Matrix.html) (accessed Nov. 03, 2021).

# Appendices

# Appendix A Laser processing parameters

Table A.1 Parameters used to laser process the sample shown in Figure 3.7.

<b>Square Name</b>	<b>Scanning speed [mm s<sup>-1</sup>]</b>	<b>Repetition rate [kHz]</b>	<b>Polarization [°]</b>	<b>Fluence [mJ cm<sup>-2</sup>]</b>	<b>Interline spacing [μm]</b>
A1	2 000	300	0°	413	10
A2	2 000	300	0°	69	10
A3	125	300	0°	413	10
A4	125	300	0°	69	10
A5	15	300	0°	151	10
A6	5	300	0°	69	10
B1	2 000	300	0°	850	10
B2	2 000	300	0°	151	10
B3	300	300	0°	413	10
B4	125	300	0°	151	10
B5	15	300	0°	413	10
B6	15	300	0°	69	10

Table A.1 Parameters used to laser process the sample shown in Figure 4.10

<b>MS n°</b>	<b>Scanning speed [mm s<sup>-1</sup>]</b>	<b>Repetition rate [kHz]</b>	<b>Number of passes</b>	<b>Polarization [°]</b>	<b>Fluence [mJ cm<sup>-2</sup>]</b>	<b>Interline spacing [μm]</b>
MS1	100	600	3	0	1 641	10
MS2	500	600	6	0	372	10
MS3	200	600	1	0	274	10
MS4	200	600	1	0	116	10
MS5	500	600	6	0	29	10
MS6	200	600	6	0	466	10

Table A.2 Laser processing parameter sets used to produce the samples of Chapter 5.

<i>Figure 5.9</i>	MS n°	Scanning speed [mm s <sup>-1</sup> ]	Repetition rate [kHz]	Polarization [°]	Fluence [mJ cm <sup>-2</sup> ]	Interline spacing [μm]
	MS1	220	600	60	149	10
	MS2	220	600	0	149	10
	MS3	220	600	0	466	10
	MS4	220	600	60	466	10
<i>Figure 5.10</i>	MS n°	Pulse number	Repetition rate [kHz]	Fluence [mJ cm <sup>-2</sup> ]		
	MS1	15	300	221		
	MS2	450	300	71		
	MS3	15	300	514		
	MS4	45	600	221		

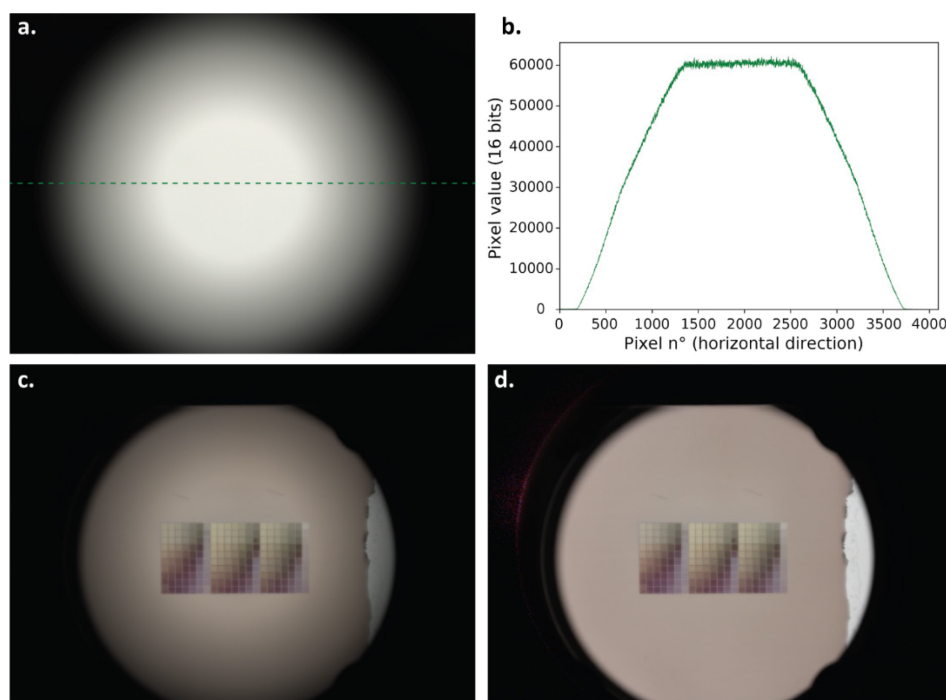
Figure 5.11	MS n°	Scanning speed [mm s <sup>-1</sup> ]	Repetition rate [kHz]	Polarization [°]	Fluence [mJ cm <sup>-2</sup> ]	Interline spacing [μm]
	MS1	50	600	90	108	10
	MS2	150	600	0	273	10
	MS3	250	600	0	273	10
	MS4	50	300	90	512	10
	MS5	5	300	90	227	10
	MS6	5	300	0	184	10
	MS7	250	300	0	512	10
	MS8	200	300	90	629	10
Figure 5.12	MS n°	Scanning speed [mm s <sup>-1</sup> ]	Repetition rate [kHz]	Polarization [°]	Fluence [mJ cm <sup>-2</sup> ]	Interline spacing [μm]
	MS1	90	600	30	70	10
	MS2	90	600	60	70	10
	MS3	90	600	120	70	10
	MS4	90	600	150	70	10

	MS5	120	600	30	70	10	
	MS6	120	600	60	70	10	
	MS7	120	600	120	70	10	
	MS8	120	600	150	70	10	
<i>Figure 5.13</i>	MS n°	Scanning speed [mm s <sup>-1</sup> ]	Repetition rate [kHz]	Number of passes	Polarization [°]	Fluence [mJ cm <sup>-2</sup> ]	Interline spacing [μm]
	MS1	100	600	3	0	1 641	10
	MS2	100	600	3	90	1 641	10
	MS3	500	600	6	0	372	10
	MS4	200	600	1	0	274	10
	MS5	200	600	1	0	116	10
	MS6	500	600	6	0	29	10
	MS7	200	600	6	0	466	10
	MS8	200	600	6	90	466	10

## Appendix B Image normalization

### Image normalization

One drawback of the telecentric lens is that its transmission decreases rapidly when moving away from the optical axis. To correct the inhomogeneous radiometry of the telecentric lens, the images taken with our setup are divided by an image of the light source taken in the transmission mode (Figure B.1a). A horizontal profile taken along the middle of the image (dashed green line) demonstrates the decrease of the green pixel value in the field. The same is true for red and blue pixels. This is an achromatic phenomenon, which will affect the lightness of the extracted colors if not corrected. Figure B.1c shows the picture of a sample taken in mode “transmission” with our set-up. The decay in lightness is barely perceptible by eye ; however, it can be observed on the unprocessed part of the film which transitions from light brown to a darker shade of brown on the side of the field. Note that the final transition to black is more abrupt here because of the sample rotation stage, which has a 1” circular aperture. On the contrary, in the normalized image (Figure B.1d), this lightness decay is corrected.



*Figure B.1 Demonstration of the image normalization procedure. (a) Picture of the light source taken with our set-up. (b) Profile extracted along the green dashed line in the green channel of the light source picture. (c) Picture of a sample taken with our set-up, where the center of the image field is brighter than the borders. (d) Result of the image normalization procedure.*

## Appendix C Colorimetry

This appendix describes the computation of colorimetric quantities as specified by the standards of the Commission Internationale de l'Eclairage (CIE).

### C.1 CIE 1931 XYZ Tristimulus values

The X, Y, Z tristimulus values of a light stimulus with a given spectrum  $S(\lambda)$  are usually calculated directly with the following equations:

$$\begin{aligned} X &= k \int_{380 \text{ nm}}^{750 \text{ nm}} S(\lambda) \bar{x}(\lambda) d\lambda \\ Y &= k \int_{380 \text{ nm}}^{750 \text{ nm}} S(\lambda) \bar{y}(\lambda) d\lambda \\ Z &= k \int_{380 \text{ nm}}^{750 \text{ nm}} S(\lambda) \bar{z}(\lambda) d\lambda \end{aligned} \quad (\text{C.1})$$

where the normalization factor  $k$  is  $K_m = 683 \text{ lm/W}$ . The X, Y, and Z tristimulus values are all positive. The color matching function  $\bar{y}(\lambda)$  coincides with the sensitivity function of the human visual system used in photometry,  $V(\lambda)$ . The CIE recommends for  $S(\lambda)$  a spectral resolution of 5 nm or less[122].

The tristimulus values can also be computed for a non-luminous object, from the spectrum of light reflected or transmitted by the object with spectral reflectance  $R(\lambda)$  or  $T(\lambda)$ , under a certain lighting with spectral distribution power  $S(\lambda)$ . In this case, they are given by

$$\begin{aligned} X &= k \int_{380 \text{ nm}}^{750 \text{ nm}} R(\lambda) S(\lambda) \bar{x}(\lambda) d\lambda \\ Y &= k \int_{380 \text{ nm}}^{750 \text{ nm}} R(\lambda) S(\lambda) \bar{y}(\lambda) d\lambda \\ Z &= k \int_{380 \text{ nm}}^{750 \text{ nm}} R(\lambda) S(\lambda) \bar{z}(\lambda) d\lambda \end{aligned} \quad (\text{C.2})$$

with

$$k = \frac{100}{\int_{380 \text{ nm}}^{750 \text{ nm}} S(\lambda) \bar{y}(\lambda) d\lambda} \quad (\text{C.3})$$

For a perfect white reflector ( $R = 1$  for all visible wavelengths) or a perfect transparent object ( $T = 1$ ), the Y value is 100.

Since the XYZ tristimulus values are calculated from spectra by using linear operators, the tristimulus values of the mixing two primary colors is a weighted average of their tristimulus values.



## C.2 Illuminants

Standard light sources have been defined by the Commission Internationale de l'Eclairage (CIE) in order to ease the communication worldwide on color appearance of materials under different lightings. These standard spectral power distributions (SPDs) are called *illuminants* [122]. The illuminant that would have uniform relative SPD is called *equal energy illuminant, E*. It is theoretical and not feasible in practice with artificial light sources. Other illuminants are inspired of the SPDs of incandescent light (illuminant A), daylight (D illuminants series), fluorescent lightings (illuminants F), and LEDs. Some of them are plotted in Figure C.1.

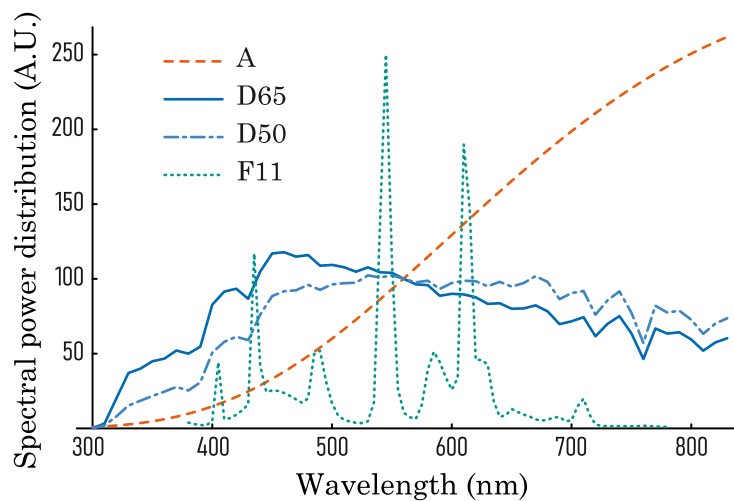


Figure C.1 Relative spectral power distributions of CIE standard illuminants A, D65, D50 and F11.

The D series of illuminants were constructed by Judd, MacAdam, and Wyszecki to represent natural daylight. The D50 and D65 illuminants are the main ones used in the graphical industry and paper industry. Their spectra, plotted in Figure C.1, correspond respectively to horizon daylight and noon daylight spectra.

## C.3 Standard RGB color space (sRGB)

In the case where color has to be measured by a color camera or displayed on a monitor or television screen, three values or *channels* are usually used to encode the color of each pixel. In the case of red green blue (RGB) images, these values are intended to be the lightnesses of three colored light sources which are mixed together. The mixing is assumed to be additive. These light sources are called primaries, and can correspond, for example, to the three light sources used in the computer screen that will display the image. Taken together, they define a color space in which all colors that can be produced by mixing these three primaries can be expressed in. These colors form its *gamut*. When the color primaries are the one of a specific display device, the color space is said to be

*device-dependent*. However, in practice, due to the different technologies of light sources and color filters available, the color of the primaries varies greatly from one display to another, which makes it impossible to code the images for each specific kind of display. For this reason, standard color spaces were developed, in which the colors of the primaries are fixed to tabulated values in a *device-independent* space such as the CIE 1931 XYZ color space. This is the case of the standard red green blue (sRGB) space defined by the American Society for Testing and Materials (ASTM)[123].

As the primaries defining the sRGB space are defined in the CIE 1931 XYZ color space, sRGB values represent linear mixings of these three can be converted to XYZ tristimulus values by a matrix multiplication with a standard conversion matrix M:

$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = M \begin{bmatrix} R \\ G \\ B \end{bmatrix} \quad (\text{C.4})$$

When the white point is set to D65, the matrix M is defined as follows[124]:

$$M = \begin{bmatrix} 0.4124564 & 0.3575761 & 0.1804375 \\ 0.2126729 & 0.7151522 & 0.0721750 \\ 0.0193339 & 0.1191920 & 0.9503041 \end{bmatrix}$$

#### C.4 CIE 1976 L\*a\*b\* color space and color appearance models

The CIE 1976 L\*a\*b\* model is a three-dimensional color representation system where the  $L^*$ ,  $a^*$ , and  $b^*$  coordinates are calculated from the CIE 1931 tristimulus values  $X, Y, Z$  of the considered color and  $X_n, Y_n, Z_n$ , the tristimulus values of a color defined as reference white for the chromatic adaptation:

$$\begin{aligned} L^* &= 116f(Y/Y_n) - 16 \\ a^* &= 500[f(X/X_n) - f(Y/Y_n)] \\ b^* &= 200[f(Y/Y_n) - f(Z/Z_n)] \end{aligned} \quad (\text{C.5})$$

with

$$f(q) = \begin{cases} q^{1/3} & \text{if } q > 0.008856 \\ 7.787q + \frac{16}{116} & \text{if } q \leq 0.008856 \end{cases} \quad (\text{C.6})$$

The  $L^*$  axis corresponds to the *lightness*, between 0 (black) and 100 (white). The other two coordinates are used to describe the *hue* and *chroma*. The following quantities serve as correlates of chroma  $C_{ab}^*$  and hue  $h_{ab}^*$ :

$$C_{ab}^* = \sqrt{a^{*2} + b^{*2}}; \quad h_{ab}^* = \arctan(b^* / a^*) \quad (\text{C.7})$$

The further away from the  $L^*$  axis, the higher the chroma (brighter colors); on the contrary, the  $L^*$  axis contains the achromatic grays. This space is one of the first *color appearance models*, and it is by far the most widely used. It is considered *perceptually uniform*, which means there is a correlation between distances in this space and perceptual distances between colors.

Beyond considering the chromatic adaptation (although imperfectly), the CIE 1976  $L^*a^*b^*$  model enables estimating the perceptual difference between two colors  $(L_1^*, a_1^*, b_1^*)$  and  $(L_2^*, a_2^*, b_2^*)$  with metrics, denoted  $\Delta E$ . The original 1976 metric, usually denoted by  $\Delta E_{ab}^*$ , is simply the Euclidean distance between the points representing the colors to be compared:

$$\Delta E_{ab}^* = \sqrt{(\Delta L^*)^2 + (\Delta a^*)^2 + (\Delta b^*)^2} \quad (\text{C.8})$$

$$\text{with } \Delta L^* = L_1^* - L_2^*, \Delta a^* = a_1^* - a_2^*, \Delta b^* = b_1^* - b_2^*.$$

Note that each of these differences of  $L^*$ ,  $a^*$  and  $b^*$  can also be considered independently. In the present work, in particular,  $\Delta L^*$  is used to express the luminance contrast between two colors.

The *just noticeable difference (JND)* between colors, i.e., the value below which two different colors look identical for most people, has been established around 2.2 units for a non-trained observer.

A more reliable metric was then proposed in 1994:

$$\Delta E_{94}^* = \sqrt{\left(\frac{\Delta L^*}{k_L S_L}\right)^2 + \left(\frac{\Delta C_{ab}^*}{k_C S_C}\right)^2 + \left(\frac{\Delta H_{ab}^*}{k_H S_H}\right)^2}, \quad (\text{C.9})$$

with

$$\begin{aligned} \Delta C_{ab}^* &= C_1^* - C_2^* \\ C_i' &= \sqrt{a_i'^2 + b_i'^2} \quad i=1,2 \\ \Delta H_{ab}^* &= \sqrt{(\Delta a^*)^2 + (\Delta b^*)^2 - (\Delta C_{ab}^*)^2} \end{aligned} \quad (\text{C.10})$$

and where  $S_L = 1$ ,  $S_C = 1 + K_2 C_1^*$ ,  $S_C = 1 + K_1 C_1^*$ ,  $k_L$ ,  $K_1$  and  $K_2$  are constants depending on the application domain. For example, in the graphical arts industry, one considers:  $k_L = 1$ ,  $K_1 = 0.045$  and  $K_2 = 0.015$ , whereas in the textile industry, one rather considers:  $k_L = 2$ ,  $K_1 = 0.048$ ,  $K_2 = 0.014$ . The just noticeable difference with this metric has been established around 1 unit.

In 2000, a new metric which tends to prevail today, the CIE DE 2000, or  $\Delta E_{00}$ , was proposed to correct perceptual non-uniformity of the previous metrics. It relies on more complex formulae[82]:

$$\Delta E_{00} = \sqrt{\left(\frac{\Delta L'}{k_L S_L}\right)^2 + \left(\frac{\Delta C'}{k_C S_C}\right)^2 + \left(\frac{\Delta H'}{k_H S_H}\right)^2} + R_T \frac{\Delta C'}{k_C S_C} \frac{\Delta H'}{k_H S_H} \quad (\text{C.11})$$

with

$$L' = L^*$$

$$\Delta L' = L'_1 - L'_2$$

$$\bar{L} = (L'_1 + L'_2) / 2$$

$$\bar{C} = (C_1^* + C_2^*) / 2$$

$$a' = a^* + \frac{a^*}{2} \left( 1 - \sqrt{\frac{\bar{C}^7}{\bar{C}^7 + 25^7}} \right)$$

$$b' = b^*$$

$$C' = \sqrt{a'^2 + b'^2}$$

$$\Delta C' = C'_1 - C'_2$$

$$\bar{C}' = (C'_1 + C'_2) / 2$$

$$h' = \arctan(b' / a') \bmod(360^\circ)$$

$$\Delta h' = h'_1 - h'_2$$

$$\bar{h}' = (h'_1 + h'_2) / 2$$

$$\Delta H' = 2\sqrt{C'_1 C'_2} \sin\left(\frac{\Delta h'}{2}\right)$$

$$S_L = 1 + \frac{0.015(\bar{L} - 50)^2}{\sqrt{20 + (\bar{L} - 50)^2}}$$

$$S_C = 1 + 0.045\bar{C}'$$

$$S_H = 1 + 0.015\bar{C}'T$$

$$T = 1 - 0.17 \cos(\bar{h}' - 30) + 0.24 \cos(2\bar{h}') + 0.32 \cos(3\bar{h}' + 6) - 0.20 \cos(4\bar{h}' - 63)$$

$$R_T = -2 \sqrt{\frac{(\bar{C}')^7}{(\bar{C}')^7 + 25^7}} \sin \left[ 60^\circ \exp \left[ - \left( \frac{\bar{h}' - 275^\circ}{25^\circ} \right)^2 \right] \right]$$

$$k_L = k_C = k_H = 1$$

A space like CIE 1976 L\*a\*b\* based on a colored appearance model is suitable to characterize the color of objects. However, it was originally defined for objects that opaque and mat, and other aspects such as transparency and shine may significantly influence the perceived color.