

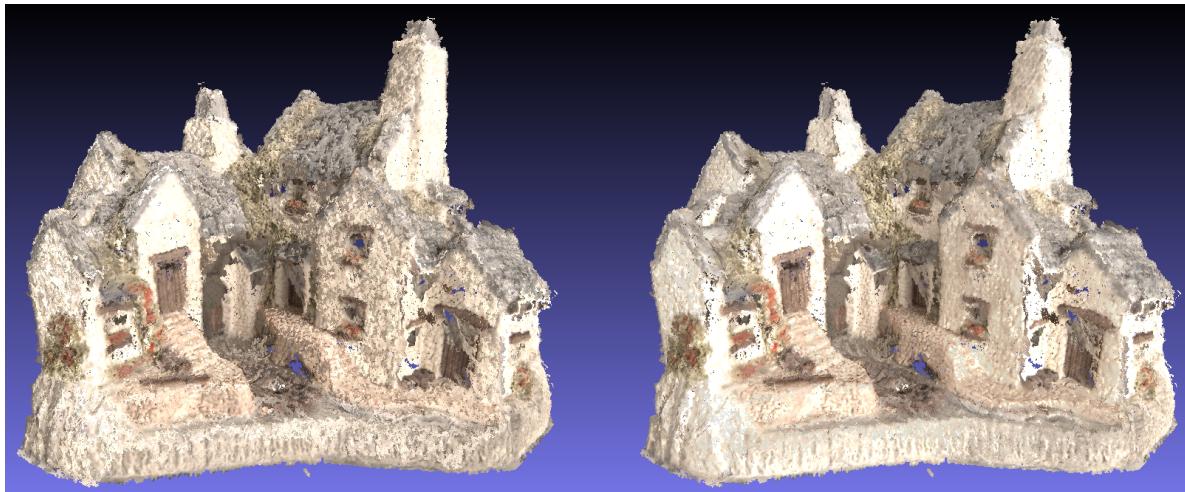
Stage de fin d'études

Cycle des Ingénieurs diplômés de l'ENSG 3<sup>ème</sup> année

---

## Égalisation radiométrique de nuages de points 3D : Principes et Algorithmique

---



**Luc Girod**

Septembre 2013

Non confidentiel    Confidential IGN    Confidential Industrie    Jusqu'au ...



## Jury

### **Président de jury :**

Olivier Dissard, IGN/ENSG

### **Commanditaire :**

Marc Pierrot-Deseilligny, IGN/ENSG/LGA

### **Rapporteur :**

Bruno Vallet, IGN/MATIS

### **Responsable pédagogique du cycle Ingénieur :**

Serge Botton, IGN/ENSG/DE/DPTS

### **Tuteur du stage pluridisciplinaire :**

Patricia Parisi, IGN/ENSG/DE/DSHI

© ENSG

## **Stage de fin d'étude du 28/04/2013 au 20/09/2013**

**Diffusion web :**  Internet  Intranet Polytechnicum  Intranet ENSG

### **Situation du document :**

Rapport de stage de fin d'études présenté en fin de 3<sup>ème</sup> année du cycle des Ingénieurs

**Nombres de pages :** 51 pages dont 3 d'annexes

**Système hôte :** L<sup>A</sup>T<sub>E</sub>X

### **Modifications :**

EDITION	REVISION	DATE	PAGES MODIFIEES
1	0	09/2013	Création



# Remerciements

---

A l'issue de ce stage, je remercie :

**Marc Pierrot-Deseilligny** Mon maître de stage, pour son aide précieuse tout au long de l'été ;

**Raphaëlle Heno et Patricia Pellardi du DIAS (ENSG) et Nicolas Martin-Beaumont du MAP** pour leur aide, conseil et fourniture de jeux de données ;

**Les membres du LGA et du projet Culture 3D** pour leur collaboration tout au long du stage ;

**Samuel Nahmani (IGN/LAREG)** pour la relecture de mon mémoire et ses commentaires constructifs ;

**Olivier Dissard** pour le prêt de la maquette de maison ayant servi à la réalisation du jeu "Maison miniature", sur lequel nombre de tests ont été basés.

## Résumé

Si les problèmes de colorimétrie dans le mosaïquage d'images ont été objet d'études approfondies par le passé et qu'ils sont maintenant globalement résolus, il n'en est pas de même pour l'égalisation des scènes non planaires et des produits photogrammétriques 3D associés. En effet, certains produits photogrammétriques ne sont pas des images mais des produits purement 3D, de type nuage de points notamment. Cependant, la cohérence colorimétrique reste d'une grande importance dans ces cas pour une visualisation agréable de résultats.

Ce rapport relate donc les recherches effectuées sur le sujet pendant l'été 2013 au Laboratoire de Géomatique Appliquée de l'IGN/ENSG et l'implémentation dans la librairie MicMac d'algorithme de correction colorimétrique appliqués aux nuages de points dont la colorimétrie provient de plusieurs images.

Deux grands points sont abordés : la correction du vignettage des images d'une part, ce défaut posant des problèmes d'homogénéité intra-image, et l'égalisation inter-images d'autre part.

**Mots-clés :** photogrammétrie, radiométrie, 3D, couleur, vignette, MicMac

## **Abstract**

If the radiometric problems in mosaicking images have been the objects of in depth research in the past and are considered solved, the counterpart of these solutions for non-planar scenes and therefore 3D photogrammetric products are still very much unexplored. These products, like 3D point clouds, are nothing like images mosaics. However, the radiometric coherence is also of great importance for these products, ensuring an optimal visualization comfort.

This report presents the research done on the subject through the summer of 2013 at the LGA (IGN/ENSG) and the implementation in the MicMac library of radiometric equalization algorithms applied to 3D point clouds were the radiometry of the points is extracted from more than one image.

Two main points are discussed. Firstly, the correction of the vignette effect, which results in a lack of homogeneity in an image. Secondly, the actual inter-images equalization.

**Keywords:** photogrammetry, radiometry, 3D, color, vignette, MicMac



# Table des matières

<b>Glossaire et sigles utiles</b>	<b>9</b>
<b>Introduction</b>	<b>11</b>
<b>1 Contexte et Objectifs</b>	<b>13</b>
1.1 Contexte d'accueil . . . . .	13
1.2 Objectifs . . . . .	13
1.3 Démarche . . . . .	14
1.3.1 Approche physique . . . . .	14
1.3.2 Approche empirique . . . . .	14
1.4 Insertion dans la chaîne de traitement . . . . .	14
<b>2 Correction d'une irrégularité radiométrique : le vignettage</b>	<b>15</b>
2.1 Origines du vignettage . . . . .	15
2.2 Modélisation du vignettage . . . . .	16
2.3 Méthode de correction . . . . .	17
2.4 Estimation du modèle . . . . .	17
2.4.1 Données en entrée . . . . .	17
2.4.2 Normalisation des données en entrée . . . . .	17
2.4.3 Mise en équation . . . . .	17
2.4.4 Résolution du système . . . . .	18
2.4.4.1 Résolution par la méthode des moindres carrés . . . . .	18
2.4.4.2 Résolution par la méthode de RANSAC . . . . .	18
2.5 Jeu test "Alarme" . . . . .	20
2.6 Conclusion . . . . .	21
<b>3 Égalisation radiométrique d'un ensemble d'images</b>	<b>23</b>
3.1 Introduction . . . . .	23
3.1.1 Cas de la mosaïque d'image, généralités . . . . .	23
3.1.1.1 Techniques présentées dans la littérature . . . . .	23
3.1.1.2 Impraticabilité de ces techniques au cas étudié . . . . .	23
3.1.2 Cas du nuage de points 3D . . . . .	24
3.2 Processus de correction . . . . .	25
3.3 Algorithme d'extraction des points de liaison radiométrique . . . . .	25
3.3.1 Présentation de l'algorithme . . . . .	25
3.3.2 Limites du concept . . . . .	28
3.4 Choix du modèle . . . . .	28
3.4.1 Modèle multiplicatif . . . . .	28
3.4.2 Modèle polynômial . . . . .	28
3.4.3 Une approche plus locale, le modèle interpolé . . . . .	30
3.4.3.1 Justification de l'approche . . . . .	30
3.4.3.2 Algorithme . . . . .	30

## 4 TABLE DES MATIÈRES

---

3.4.3.3	Fonctionnement pour plus d'une image . . . . .	30
3.5	Analyse des résultats . . . . .	32
3.5.1	Présentation des jeux test . . . . .	32
3.5.1.1	Jeu "Baigneuse Maillo" . . . . .	32
3.5.1.2	Jeu "Maison Miniature" . . . . .	33
3.5.2	Résultats du processus d'égalisation . . . . .	35
3.5.3	Influences des paramètres des algorithmes . . . . .	37
3.5.3.1	Influence de la résolution . . . . .	37
3.5.3.2	Influence de la définition de proximité . . . . .	38
3.5.3.3	Influence de la correction du vignettage . . . . .	39
3.6	Conclusion : améliorations possibles et pistes de recherche . . . . .	40
3.6.1	Estimation du BRDF . . . . .	40
3.6.2	Blending 3D . . . . .	40
<b>4</b>	<b>Implémentation des algorithmes</b>	<b>41</b>
4.1	Prise en main des méthodes de développement . . . . .	41
4.1.1	Tests sous Octave . . . . .	41
4.1.2	La librairie MicMac . . . . .	41
4.1.3	Développement coopératif avec Mercurial . . . . .	41
4.2	V.O.D.K.A. . . . .	42
4.2.1	Nom et fonction . . . . .	42
4.2.2	Données en entrée . . . . .	42
4.2.3	Gestion des jeux d'images . . . . .	42
4.2.4	Données en sortie . . . . .	42
4.2.5	Options . . . . .	42
4.2.6	Temps de traitement . . . . .	42
4.3	A.R.S.E.N.I.C . . . . .	43
4.3.1	Nom et fonction . . . . .	43
4.3.2	Données en entrée . . . . .	43
4.3.3	Données en sortie . . . . .	43
4.3.4	Options . . . . .	43
4.3.5	Temps de traitement . . . . .	43
4.4	Parallélisation . . . . .	44
4.5	Conclusion . . . . .	44
<b>Conclusion</b>	<b>45</b>	
<b>Bibliographie</b>	<b>47</b>	
<b>A Documentation et code source publié</b>	<b>51</b>	

# Table des figures

1	Fusion de plusieurs nuages de points avec une ligne de démarcation visible . . . . .	11
1.1	Bannière de fin d'execution de MicMac . . . . .	13
1.2	Insertion des commandes ans la chaine de traitement . . . . .	14
2.1	Origine géométrique du vignettage . . . . .	15
2.2	Vignette du Nikkor 18-105 mm VR, F/3.6,18mm sur un champ gris homogène . . . . .	16
2.3	Mauvaise correction du vignettage obtenue par moindre carré - jeu "Alarme" . . . . .	18
2.4	Bonne correction du vignettage obtenue par la méthode de RANSAC - jeu "Alarme" . . . . .	19
2.5	Image de champs gris homogène corrigée du vignettage . . . . .	20
2.6	7 images du jeu "Alarm" . . . . .	20
2.7	Image centrale du jeu "Alarme" avant et après correction du vignettage . . . . .	21
3.1	Apollo 11. De haut en bas : images originales ; traitement par blending ; correction des vignettes; vignettes + blending. In <i>Vignette and Exposure Calibration and Compensation</i> , [3] . . . . .	24
3.2	Diagramme des étapes du processus d'égalisation . . . . .	25
3.3	Diagramme de l'algorithme de détection des points de liaison . . . . .	26
3.4	Schéma de l'algorithme de détection des points de liaison . . . . .	27
3.5	331 points de liaisons extrait de l'image 2 du jeu "Maison Miniature" . . . . .	27
3.6	Fusions de nuages égalisés "en noir" . . . . .	29
3.7	Images mal corrigées : image noircie (gauche) et brûlée (droite) . . . . .	29
3.8	Surface de correction pour l'image 3 du jeu "Maison Miniature" . . . . .	31
3.9	Surface de correction pour l'image 2 du jeu "Maison Miniature" . . . . .	31
3.10	5 images principales du jeu "Baigneuse Maillol" - ©MAP . . . . .	32
3.11	"AperiCloud" du jeu "Baigneuse Maillol" - ©MAP . . . . .	33
3.12	5 images principales du jeu "Maison Miniature" - ©Luc Girod . . . . .	33
3.13	"AperiCloud" du jeu "Maison Miniature" - ©Luc Girod . . . . .	34
3.14	Nuage original (gauche) et corrigé (droite) - Maison Miniature . . . . .	35
3.15	Nuage original (gauche) et corrigé (droite) - Baigneuse Maillol . . . . .	35
3.16	Répartition des 1868 facteurs $G_j/G_i$ - Maison Miniature . . . . .	36
3.17	Répartition des 3200 facteurs $G_j/G_i$ - Baigneuse de Maillol . . . . .	37
3.18	Répartition des facteurs $G_j/G_i$ en fonction de la résolution - Maison Miniature . . . . .	37
3.19	Répartition des facteurs $G_j/G_i$ en fonction de la proximité - Maison Miniature . . . . .	38
3.20	Répartition des facteurs $G_j/G_i$ avec et sans dévignettage - Maison Miniature . . . . .	39
3.21	Définitions BRDF d'un point ©David McAllister, NVIDIA . . . . .	40



# Liste des tableaux

3.1	Statistiques du jeu "Maison Miniature" . . . . .	36
3.2	Statistiques du jeu "Baigneuse Maillol" . . . . .	36
3.3	Statistiques du jeu "Maison Miniature" en fonction de la résolution . . . . .	38
3.4	Statistiques du jeu "Maison Miniature" en fonction de la définition de la proximité . . . . .	38
3.5	Statistiques du jeu "Maison Miniature" avec et sans correction du vignettage . . . . .	39



# Glossaire et sigles utiles

---

**ENSG** École Nationale des Sciences Géographiques - Champs-sur-Marne

**IGN** Institut National de l'Information Géographique et Forestière - Saint Mandé

**LGA** Laboratoire de Géomatique Appliquée - Champs-sur-Marne

**MAP** Laboratoire de Modèles et simulations pour l'Architecture et le Patrimoine - Marseille

**RMN** Réunion des Musées Nationaux

**SIFT** Scale Invariant Feature Transform, un algorithme de détection de points de liaison

**RANSAC** RANdom SAmple Consensus, un algorithme d'estimation

**BRDF** Bidirectional Reflectance Distribution Function, une fonction physique de la réflectivité des objets



# Introduction

---

L'esthétisme. Voilà une notion à la fois centrale et très périphérique en photogrammétrie. En effet, ce domaine a longtemps eu pour objectif la seule vérité géométrique, permettant la cartographie. Alors qu'apparaît la notion de mosaïque d'images, l'importance des corrections radiométriques se fait sentir : les variations de couleurs présentes d'une image à une autre sont une véritable gêne. La photogrammétrie va alors s'intéresser à l'homogénéisation des couleurs pour produire des orthoimages esthétiquement agréables. Les techniques actuellement utilisées de manière opérationnelle proposent des résultats très convaincants pour la création de mosaïques d'images.

L'objet de ce mémoire est d'élargir la notion d'égalisation radiométrique pour des produits photogrammétriques tridimensionnels. En effet, lors de la fusion de nuages de points représentant différentes parties d'un même objet (quel qu'en soit sa nature), la couleur des points est issue pour chaque nuage d'une image différente. Cela provoque la superposition de points représentant la même zone géométriquement mais ne partageant pas la même radiométrie, ce qui est fort peu esthétique et peut rendre problématique l'interprétation du modèle.

L'importance d'une colorimétrie agréable à l'œil est d'autant plus grande que l'objectif des modèles 3D produits par photogrammétrie est souvent lié à la visualisation et l'analyse de l'objet, l'œuvre ou le bâtiment photographié. La présence d'irrégularités colorimétriques provoque une gêne certaine pour ces usages et leur correction n'avait jusqu'alors été que peu étudiée.

On approchera ici dans un premier temps les corrections des défauts d'homogénéité intrinsèques des images, notamment le problème du vignettage. On s'intéressera ensuite à l'égalisation des groupes d'images destinés à colorer des nuages de points dont la fusion représente une scène complète.

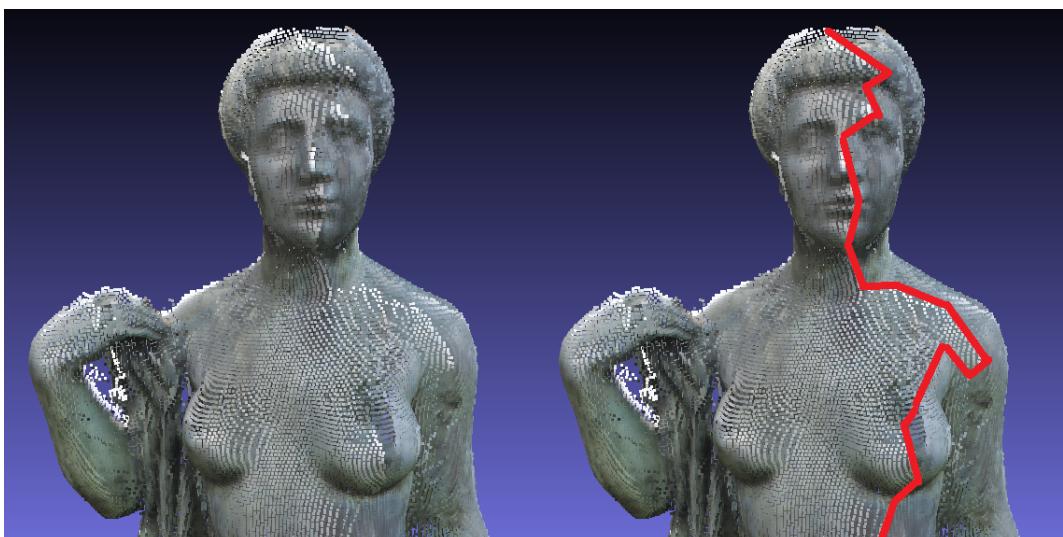


FIGURE 1 – Fusion de plusieurs nuages de points avec une ligne de démarcation visible



# CONTEXTE ET OBJECTIFS

## 1.1 Contexte d'accueil

Ce stage a été effectué du 29 avril au 20 septembre 2013 dans le Laboratoire de Géomatique Appliquée (LGA), dirigé par Marc Pierrot-Deseilligny. Ce laboratoire est l'entité de recherche associée à l'École Nationale des Sciences Géographiques (ENSG) à Champs-sur-Marne, elle-même une direction de l'Institut National de l'Information Géographique et Forestière (IGN). Les travaux actuellement menés par le laboratoire exploitent le logiciel MicMac.

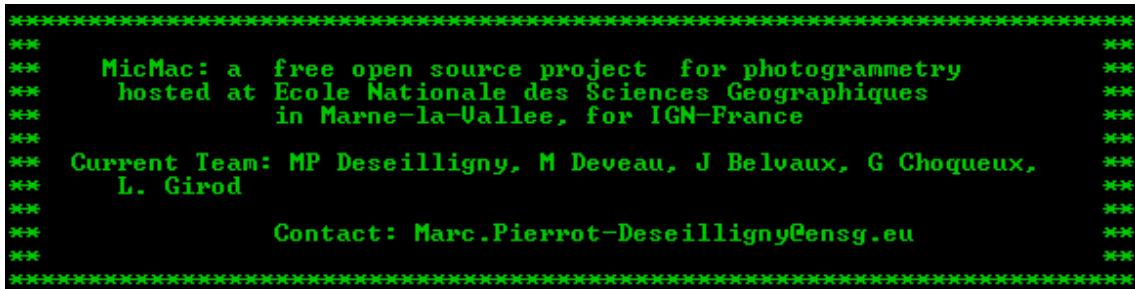


FIGURE 1.1 – Bannière de fin d'exécution de MicMac

**MicMac** est un logiciel développé dans le laboratoire d'accueil et par le groupe de travail du projet *Culture 3D* à l'IGN. Il présente une chaîne de traitements photogrammétriques complète allant du calcul des points de liaison à la corrélation dense en passant par l'estimation des paramètres de caméra et des orientations relatives et absolues, le tout de façon de plus en plus automatique au fur et à mesure de l'évolution du logiciel. Une vision très générale de la chaîne de traitement (avec le nom des commandes associées) est présenté sur le schéma de la figure 1.2.

On peut trouver le logiciel, des jeux test et sa documentation sur le site internet suivant : <http://logiciels.ign.fr/?-Micmac,3->.

## 1.2 Objectifs

L'objectif de ce stage était de développer et d'implémenter de nouveaux algorithmes de corrections radiométriques permettant une fusion sans rupture colorimétriques visibles de plusieurs nuages de points. Ces algorithmes devront être intégrés dans la librairie MicMac.

L'objectif est donc double : il s'agit d'imaginer et de tester des algorithmes novateurs et de nouvelles approches d'une part, et d'autre part de développer un code efficace, qui sera publié avec MicMac, et fonctionnant sur toutes les plateformes pour lesquelles MicMac est conçu (Linux/MacOS/Windows).

## 1.3 Démarche

Afin d'atteindre l'objectif de développement algorithmique, une démarche en deux points a été utilisée.

### 1.3.1 Approche physique

Premièrement, on effectue une correction par modèle physique des images afin d'assurer qu'aucune irrégularité radiométrique lié à l'appareil de prise de vue ne soit la source de problèmes par la suite. Pour ce type de correction, on s'appuie sur les métadonnées disponibles : ouverture, focale, ISO, temps d'exposition pour notre étude.

Dans le cas ici présenté, c'est par le développement d'une méthode de correction du vignettage que cette problématique sera traitée.

### 1.3.2 Approche empirique

Secondement, une approche empirique est nécessaire pour le développement du système d'égalisation radiométrique inter-images. On va ici travailler sur les données de façon "aveugle", sans a priori, et chercher à homogénéiser mathématiquement les données.

## 1.4 Insertion dans la chaîne de traitement

Les deux commandes qui ont été développés s'insèrent (ou sont exploitables) après différentes étapes de la chaîne de traitement Micmac. La figure 1.2 montre ces insertions.

De plus, la commande "Vodka" peut être utilisée comme une fin en soit, si un utilisateur souhaite simplement corriger des images du vignettage.

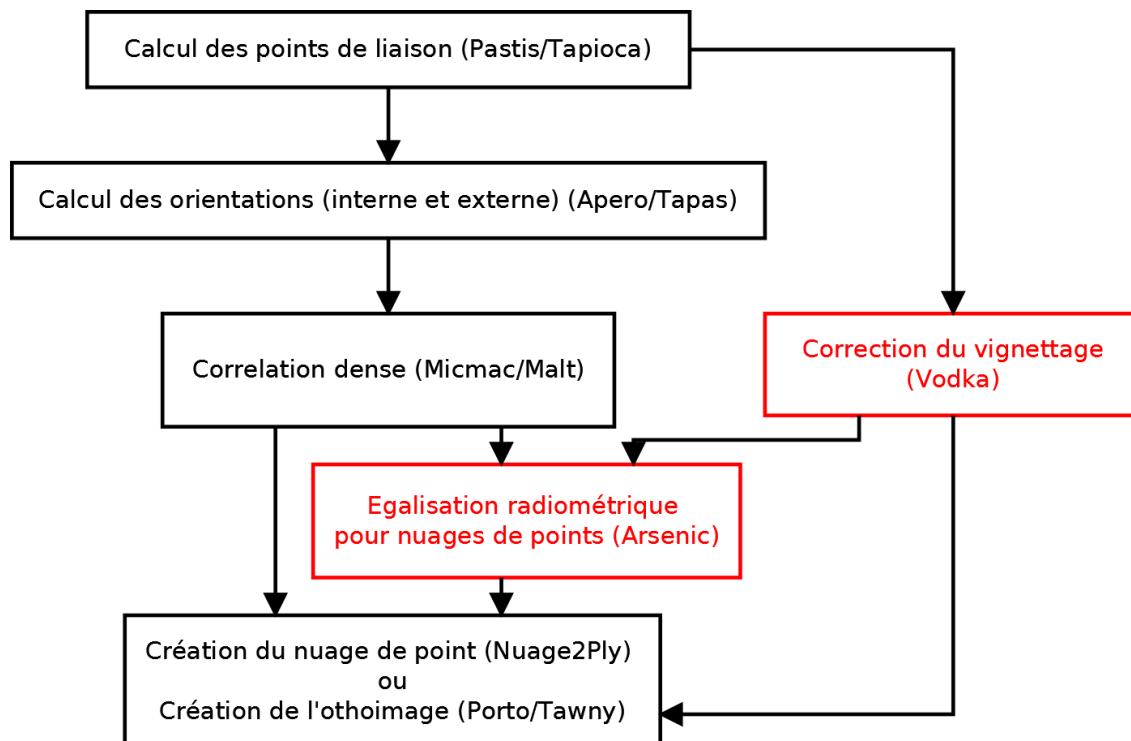


FIGURE 1.2 – Insertion des commandes ans la chaîne de traitement

# CORRECTION D'UNE IRRÉGULARITÉ : LE VIGNETTAGE

Les images produites par les appareils photographiques présentent de nombreux défauts dus soit à l'optique, soit au capteur qui lui est associé. On rencontre notamment des défauts de distorsion, d'aberrations chromatiques et de vignettage. Si les problèmes géométriques tels que la distorsion sont bien gérés par les modèles photogramétriques et que les aberrations chromatiques sont de moins en moins visibles sur les appareils modernes, le vignettage est encore bien présent. Certains photographes utilisent même cet effet à des fins artistiques, le centre de l'image étant particulièrement mis en valeur.

Il est bien sûr possible de calibrer le vignettage en laboratoire mais il dépend du paramétrage de l'optique (focale et ouverture) et nécessiterait donc de multiples calibrations, ainsi qu'un matériel spécifique adapté, pour pouvoir être corrigé systématiquement.

## 2.1 Origines du vignettage

Le vignettage est un phénomène physique ayant des origines :

- géométrique, comme montré de façon exagérée dans la figure 2.1, la même surface de capteur est éclairée par un flux lumineux moins important si elle est désaxée par rapport au diaphragme.
- optique, la multiplication des lentilles dans un objectif photographique tendant à absorber les rayons lumineux d'un facteur dépendant de leur angle d'incidence, avec un minima pour les rayons parallèles à l'axe optique du fait du chemin optique plus court pour ceux-ci.
- électronique, un rayon lumineux atteignant un pixel du capteur non orthogonalement produisant un signal électrique légèrement plus faible.

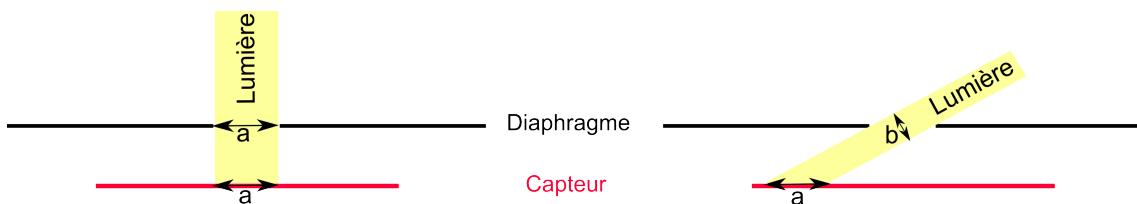


FIGURE 2.1 – Origine géométrique du vignettage

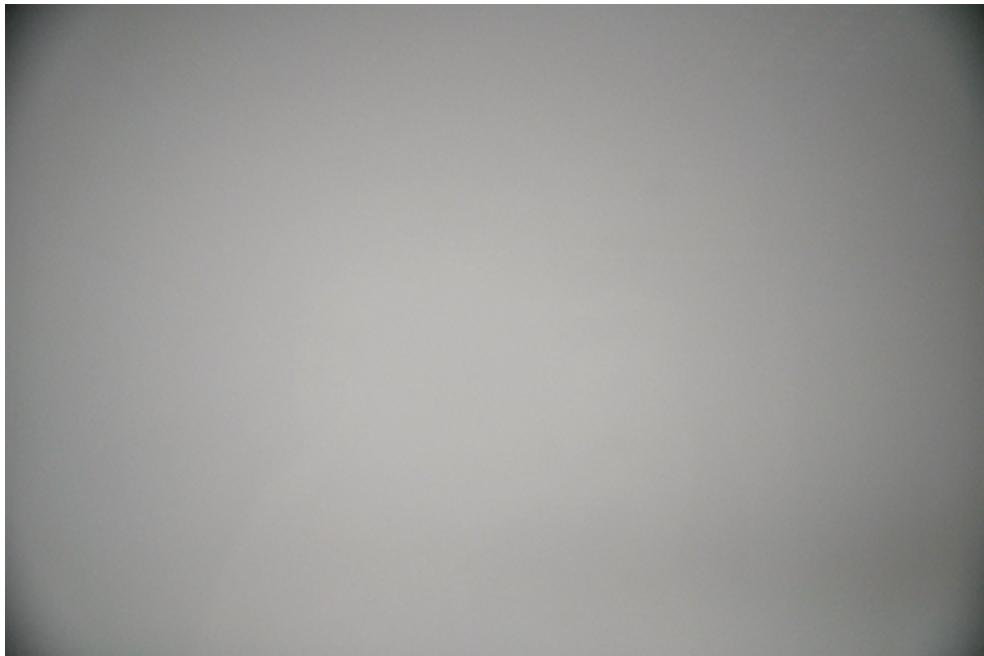


FIGURE 2.2 – Vignette du Nikkor 18-105 mm VR, F/3.6,18mm sur un champ gris homogène

Ce phénomène varie selon l'objectif utilisé, l'ouverture choisie et - dans les cas d'un objectif avec zoom - de la focale. Une forte ouverture permet le passage de rayons plus fortement écartés de l'axe optique, ce qui résulte en un effet de vignettage plus marqué. Dans le cas des zooms, l'effet est souvent plus fort aux extrêmes de la plage focale proposée, comme l'est la distorsion.

## 2.2 Modélisation du vignettage

Le phénomène de vignettage est fixe à objectif, ouverture et focale fixe. Pour corriger cet effet indésirable, il faut considérer une modélisation de la valeur enregistrée pour un pixel en fonction de la lumière perçue et des effets optiques. La valeur d'un pixel peut être exprimée par le modèle suivant, avec  $P$  la valeur du pixel,  $R$  la réponse du capteur,  $t$  le temps d'exposition,  $L$  la radianc du point dans la direction de la caméra et  $V$  la fonction de vignettage :

$$P = R(t * L * V)$$

La géométrie des optiques classiques étant conçue autour d'une symétrie axiale (l'axe optique), on sait que tout effet, comme la distorsion par exemple, sera :

- Radiale
- Infiniment dérivable
- Modélisable par un polynôme pair

La fonction de vignettage  $V$  peut donc être modélisée par une fonction polynomiale pair. L'expérience montre que le degré 6 est suffisant pour modéliser le vignettage.

On a donc la fonction suivante, avec  $r$  la distance au centre du capteur et  $\alpha$ ,  $\beta$  et  $\gamma$  les facteurs du polynôme :

$$V(r) = 1 + \alpha r^2 + \beta r^4 + \gamma r^6$$

Dans *Vignette and Exposure Calibration and Compensation* [3], Dan B. Goldman et Jin-Hung Chen démontrent que le vignettage est une fonction indépendante pouvant être estimée et corrigée en faisant abstraction de la réponse du capteur, du temps d'exposition et de la radianc des objets photographiés, créant ainsi une image simulée où :

$$P_{corrigé} = R(t * L)$$

## 2.3 Méthode de correction

Pour corriger le vignettage, on va estimer les paramètres du modèle défini en 2.2 et l'on multipliera ensuite les valeurs radiométriques de chaque point en entrée par la valeur du modèle en ce point.

## 2.4 Estimation du modèle

L'objectif de la méthode présentée ci-dessous est d'estimer un modèle de vignettage sans recourir à une calibration en laboratoire. On exploitera ici des points de liaison obtenus par l'algorithme SIFT (D.Lowe, 2004) sur un jeu d'images quelconques devant être corrigées.

### 2.4.1 Données en entrée

On dispose en entrée :

- d'un jeu d'images prises avec le même objectif, à ouverture et focale constantes
- des métadonnées des images (ISO, temps d'exposition...)
- de points de liaison extraits par l'algorithme SIFT

Chacun de ces points de liaison devrait - si les images étaient parfaites et que les objets visés avaient un comportement lambertien - avoir la même radiométrie sur chacune des images. Or le vignettage est tel qu'un point éloigné du centre de l'image sera plus sombre. En effectuant le rapport de ces niveaux d'illumination (valeur du pixel en niveau de gris), on peut donc évaluer les paramètres d'un polynôme modélisant le vignettage.

On remarquera que certaines géométries de prise de vue sont plus adaptées à l'estimation du vignettage. En effet, un point de liaison en entrée sera bien plus représentatif s'il est proche du centre d'une des images dont il est issu et plus éloigné sur une autre, mettant en avant le phénomène étudié. Une prise de vue présentant ce type de point de liaison sera donc plus adaptée. Un prise de vue convergente avec peu d'images et une petite base présentera des images dans lesquelles le vignettage est très difficilement estimable.

### 2.4.2 Normalisation des données en entrée

Lors de la modélisation du vignettage, on va considérer que la luminosité ambiante lors de la prise de vue de calibration était constante, ce qui semble raisonnable, quelques images suffisant à effectuer le calcul.

Cependant, un appareil photo en mode non manuel est susceptible d'avoir choisi des paramètres non constant. Afin d'avoir des niveaux de gris comparables, ils sont normalisés par un facteur fonction du temps d'exposition et de l'ISO de leur image d'origine et des maximums de ceux-ci dans le jeu d'image :

$$G_{enregistré} = G_{lu} * \sqrt{\frac{ExpTime * ISO}{maxExpTime * maxISO}}$$

### 2.4.3 Mise en équation

Pour chaque point de liaison, on a :

$$G_1 * (1 + \alpha r_1^2 + \beta r_1^4 + \gamma r_1^6) = G_2 * (1 + \alpha r_2^2 + \beta r_2^4 + \gamma r_2^6)$$

$\Leftrightarrow$

$$G_1 - G_2 = \alpha(G_2 * r_2^2 - G_1 * r_1^2) + \beta(G_2 * r_2^4 - G_1 * r_1^4) + \gamma(G_2 * r_2^6 - G_1 * r_1^6)$$

où  $G_i$  le niveau de gris du point dans l'image i et  $r_i$  la distance au centre pour cette image.

### 2.4.4 Résolution du système

La profusion de points de liaison mène à une très large surabondance d'équations. Trois seraient en théorie suffisante au vu du nombre d'inconnues, alors qu'on dispose de plusieurs milliers des points de liaison par couple d'images. Cette surabondance est cependant utile voire nécessaire, car elle permet d'ajuster un modèle de manière bien plus précise.

#### 2.4.4.1 Résolution par la méthode des moindres carrés

La manière la plus directe pour résoudre un tel système est l'usage de la méthode des moindres carrés. Cependant, les résultats expérimentaux tendent à démontrer le manque de robustesse de la méthode, trop influencée par les outliers générés par l'algorithme SIFT ou par le comportement non lambertien des objets présents dans l'image.

La figure 2.3 montre (par une visualisation par image 8 bits de la matrice de réels de correction) le résultat de l'estimation par moindre carré de la vignette présente dans le jeu de données "Alarme" (présenté en 2.5). On remarque immédiatement que le résultat n'a rien à voir avec ce qui serait attendu, sur-éclairant de façon majeure (et injustifiée) un anneau de l'image.

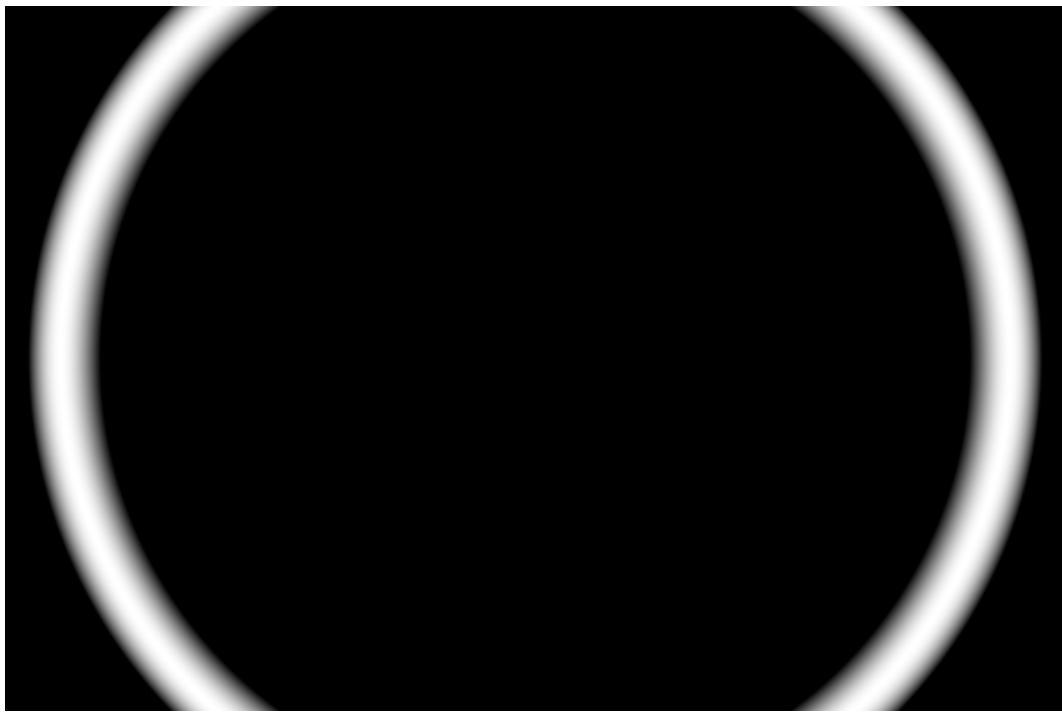


FIGURE 2.3 – Mauvaise correction du vignettage obtenue par moindre carré - jeu "Alarme"

#### 2.4.4.2 Résolution par la méthode de RANSAC

La méthode RANSAC (RANdom SAmple Consensus) utilise un sous ensemble aléatoire d'observations pour calculer les paramètres du modèle, puis confronte ce modèle au reste des données. Ce processus est itéré un grand nombre de fois et un système de score permet de déterminer le meilleur des modèles proposés.

Dans l'implémentation utilisée ici, l'algorithme sélectionne aléatoirement entre 9 et 30 points de liaison et calcule des paramètres du polynôme de correction par la méthode des moindres carrés. Les paramètres de sortie sont ensuite testés pour vérifier leur cohérence avec un modèle probable. En effet, un modèle cohérent de vignette présentera toujours un  $\alpha > 0$ .

Si un modèle estimé correspond à ces critères, on calcule alors :

- le pourcentage  $P_{inliers}$  des points en accord avec ce modèle à  $\pm 5$  niveau de gris (sur un échantillonnage sur 256 niveau), ce choix étant empirique et lié à l'approximation considérée comme acceptable du modèle de correction  $\Rightarrow$  dans les jeux tests, environ 20% à 40% des points sont des inliers sur les meilleurs jeux de paramètres testés.
- l'erreur moyenne  $EMP$  entre le modèle et les points, pondérée par le minimum de  $[r_1, r_2]$ , mettant ainsi en avant les points plus proches des bords plus affectés par le vignettage.

On calcule alors le score :

$$Score = \frac{P_{inliers}}{EMP}$$

Cette opération est répétée 10000 fois au minimum, et se poursuit jusqu'à ce que 500 modèles aient passé le test de cohérence. A chaque fois qu'un modèle dépasse le modèle de score maximum enregistré, il remplace celui-ci. Ces seuils, très empiriques, sont le produits de tests et un compromis entre la vitesse de calcul et l'espérance d'obtention d'un résultat de qualité, cette dernière étant prioritaire.

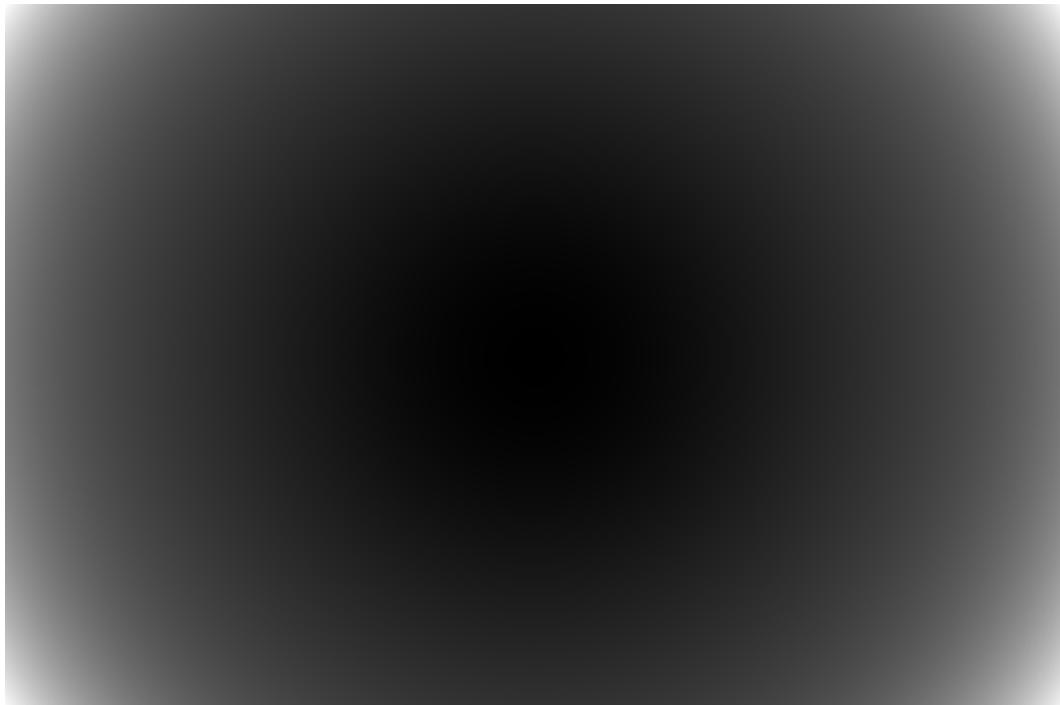


FIGURE 2.4 – Bonne correction du vignettage obtenue par la méthode de RANSAC - jeu "Alarme"

La figure 2.4 est l'équivalent de la figure 2.3 pour une estimation par l'algorithme de RANSAC présenté ici. L'apparence est cohérente avec une vignette, on verra en 2.5 et sur la figure 2.5 que son application est satisfaisante.

La figure 2.5 est l'image de la figure 2.2 après correction du vignettage avec la vignette estimée par le jeu "Alarme" présenté en 2.5. On remarque que l'image est très homogène, ne présentant que 20 niveaux de gris, contre plus de 70 pour l'image originale. Le vignettage n'est ici plus visible et l'on peut alors véritablement considérer le problème comme étant corrigé, malgré la géométrie de prise de vue difficile.



FIGURE 2.5 – Image de champs gris homogène corrigée du vignettage

## 2.5 Jeu test "Alarme"

Pour tester les algorithmes, un jeu d'images convergentes, et donc potentiellement problématique a été réalisé (cf. figure 2.6). Le sujet est le déclencheur d'alarme incendie situé à côté de l'ascenseur du bâtiment Cassini de l'ENSG, au quatrième étage.



FIGURE 2.6 – 7 images du jeu "Alarm"

La figure 2.7 montre l'image centrale originale et l'image en sortie de l'algorithme. On remarque bien que l'image corrigée est bien plus homogène que l'image initiale, fortement vignetée (avec un facteur 1,3 au niveau des coins).



FIGURE 2.7 – Image centrale du jeu "Alarme" avant et après correction du vignettage

## 2.6 Conclusion

On a vu dans ce chapitre les origines de l'effet de vignettage et une modélisation paramétrique du phénomène permettant sa correction. On a ensuite vu un algorithme estimant, avec une précision suffisante pour les besoins exprimés dans notre contexte, ces paramètres pour des jeux d'images variés. Nous verrons au chapitre 4 l'implémentation de cet algorithme dans la librairie MicMac.



# ÉGALISATION RADIOMÉTRIQUE D'UN ENSEMBLE D'IMAGES

CHAPITRE  
**3**

## 3.1 Introduction

Une fois le vignettage corrigé, on peut considérer que les images ne présentent plus d'irrégularités en elles-mêmes. Il convient donc alors de réaliser une égalisation radiométrique du groupe d'images qui nous intéresse pour obtenir des produits photogrammétriques dans lesquelles les zones de passage entre nuages de points provenant d'images différentes ne soit pas des zones de ruptures.

### 3.1.1 Cas de la mosaïque d'image, généralités

La littérature traitant d'égalisation radiométrique explore en général la problématique de la mosaïque d'image. Il est en effet primordial de bien assurer une cohérence radiométrique pour permettre aux produits finaux d'avoir un esthétisme correct. Parmi ces produits, on remarquera particulièrement les images panoramiques (ainsi que les variantes 360° et sphériques) et les orthoimages.

#### 3.1.1.1 Techniques présentées dans la littérature

La correction des problèmes intrinsèques aux images est traitée ici de manière similaire à ce qui a été présenté au chapitre 2, avec l'ajout notable de la correction du filtre atmosphérique pour les images aériennes et spatiales.

Les corrections inter-images, permettant la création de mosaïques, se basent soit sur différentes techniques :

- des corrections statistiques qui homogénéisent toutes les images avec en général une perte importante de dynamique
- des techniques de fusion, de type blending, qui permettent un passage doux d'une image à une autre par un floutage progressif
- des corrections dites "physiques" qui utilisent des informations sur la prise de vue pour appliquer des modèles empiriques (voir l'article de [1] dans la bibliographie)

#### 3.1.1.2 Impraticabilité de ces techniques au cas étudié

Ici, on ne cherche pas à coller les images entre elles et qu'on ignore les paramètres permettant les corrections physiques (heure précise de prise de vue, position du soleil, position et orientation de la camera en coordonnées absolues...). Les types de corrections citées précédemment ne sont donc pas applicable aux modèles tridimensionnels, ou nécessiteraient des modifications et adaptations importantes.

Le blending par exemple ne peut pas s'appliquer directement aux images. On pourrait cependant imaginer un blending de nuages de points, par une recherche des points proches et un système d'égalisation adapté. Cependant, cette approche semble complexe et il a été choisi de ne pas tester ce type de correction.

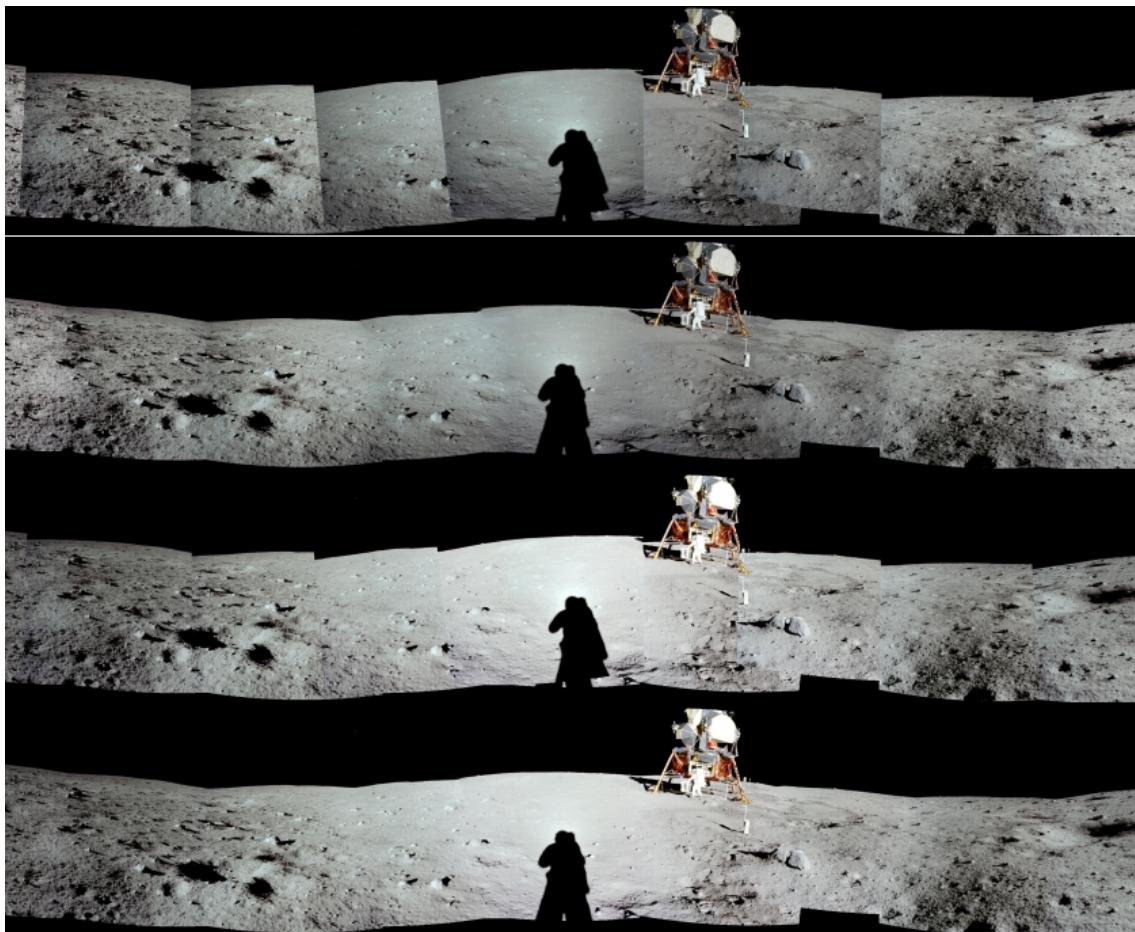


FIGURE 3.1 – Apollo 11. De haut en bas : images originales ; traitement par blending ; correction des vignettes ; vignettes + blending. In *Vignette and Exposure Calibration and Compensation*, [3]

### 3.1.2 Cas du nuage de points 3D

Pour obtenir un ensemble de nuages de points 3D offrant une cohérence radiométrique dans les zones de recouvrement, il convient de corriger chaque image utilisée pour la coloration d'un des sous-nuages.

Les différences constatées peuvent être liées à de nombreux facteurs :

- variation des paramètres de l'appareil photo (température de couleur, ouverture, ISO, temps d'exposition)
- variation de l'éclairage du sujet (naturelle ou artificielle)
- aspect non lambertien des matériaux photographiés

## 3.2 Processus de correction

Le processus d'égalisation se découpe en trois étapes principales, comme le montre le diagramme de la figure 3.2.

A partir des points de liaison radiométriques obtenus par l'algorithme présenté en 3.3, il convient de définir une matrice de correction de l'image sous forme de facteurs multiplicatifs d'égalisation pour chaque pixel de chaque image (et pour tous les canaux présents). Les différents modèles testés sont décrits en 3.4. L'application de la correction est une étape de traitement triviale de multiplication pixel à pixel.

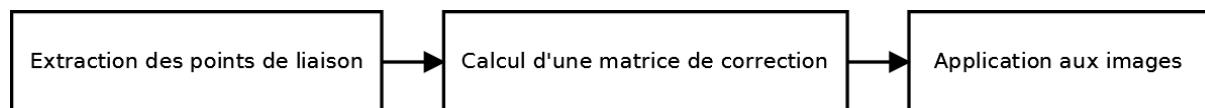


FIGURE 3.2 – Diagramme des étapes du processus d'égalisation

## 3.3 Algorithme d'extraction des points de liaison radiométrique

### 3.3.1 Présentation de l'algorithme

Pour déterminer les corrections à apporter aux images, il faut premièrement extraire des points de liaison radiométrique de celles-ci. Pour la correction du vignettage, l'utilisation des points SIFT s'était avérée pertinente et rapide - ces points étant préalablement calculés au début de la chaîne de traitement photogrammétrique. Ici, ils ne sont pas adaptés. En effet, un point SIFT est un point plus clair ou plus sombre que les points de son environnement : il est donc peu représentatif de la teinte des points qui l'entourent. De plus, on ne cherche à corriger que la zone d'intérêt à partir de laquelle la couleur des points du nuage seront extraits : utiliser des points en dehors de cette zone reviendrait à affecter l'égalisation de l'image avec des phénomènes périphériques. Le décor ou le fond de la scène représentée n'a pas lieu à influer sur les corrections appliquées. Les tests effectués montraient des défauts dans les zones de l'image en frontière entre la zone d'intérêt et le fond. Celui-ci étant généralement plus stable, une tendance à modérer les corrections (ourtant nécessaires) était visible. De plus, le temps de calcul était inutilement affecté par ces points.

Afin d'obtenir une bonne densité de points de liaison dans la zone d'intérêt, y compris sur des zones d'aplats de couleurs qui contiennent les informations les plus importantes pour ces travaux, on propose une méthode qui repose sur l'utilisation de modèles 3D issus de la corrélation dense.

Pour chaque image, on va parcourir tous les pixels indiqués par le masque de corrélation associé, chercher les coordonnées 3D de ce point sur le terrain et récupérer les points correspondants dans les autres images. Les informations 3D sont extraits des cartes de profondeur associées aux images. Le schéma 3.3 montre le processus de détection des points et de filtrage.

Comme on peut le remarquer sur le schéma 3.4, la géométrie de la prise de vue pourrait résulter à la sélection de points non homologues. En effet, l'image capturée par la caméra ne représente que le premier point rencontré par la droite partant de la caméra dans une direction donnée (A pour les images 1 et 2, B pour l'image 3). Il convient donc de filtrer les points obtenus selon la position 3D recalculée pour chaque image. Si les positions des points 3D sont très proches pour au moins deux images différentes, les informations radiométrique et de positionnement seront enregistrées, rejetées sinon.

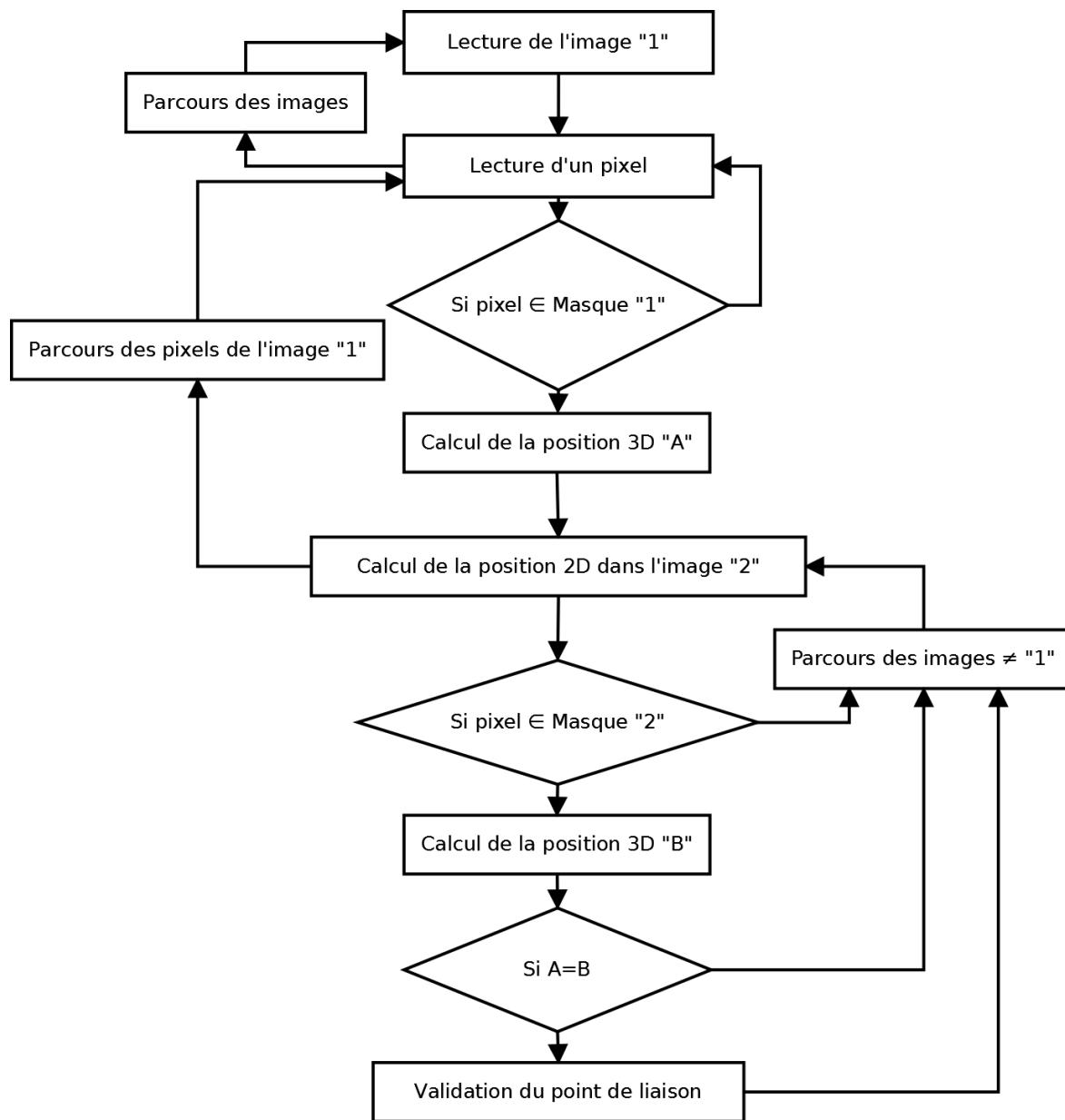


FIGURE 3.3 – Diagramme de l'algorithme de détection des points de liaison

**Définition de la notion de "proche" :** Deux positions de point seront considérées proches si la distance les séparant est inférieure à une fraction de la résolution du pixel sol, valeur variant selon le pixel et extraite de la carte de profondeur exploitée . Par l'expérience, on a choisi cette fraction égale à  $\frac{1}{16}$ . Ce seuil permet d'assurer d'une part la robustesse des points retenus et d'autre part l'extraction d'un nombre contenu de points (passer à un huitième multiplié par environ 2 le nombre de points).

On suit donc l'algorithme présenté dans le schéma de la figure 3.3 en exploitant des images sous résolues pour des raisons de vitesse de calcul, l'expérience montrant que les résultats sont satisfaisants dès une sous résolution de facteur 16.

Les résultats des expériences ayant conduit à ces choix sont présentés en 3.5.3

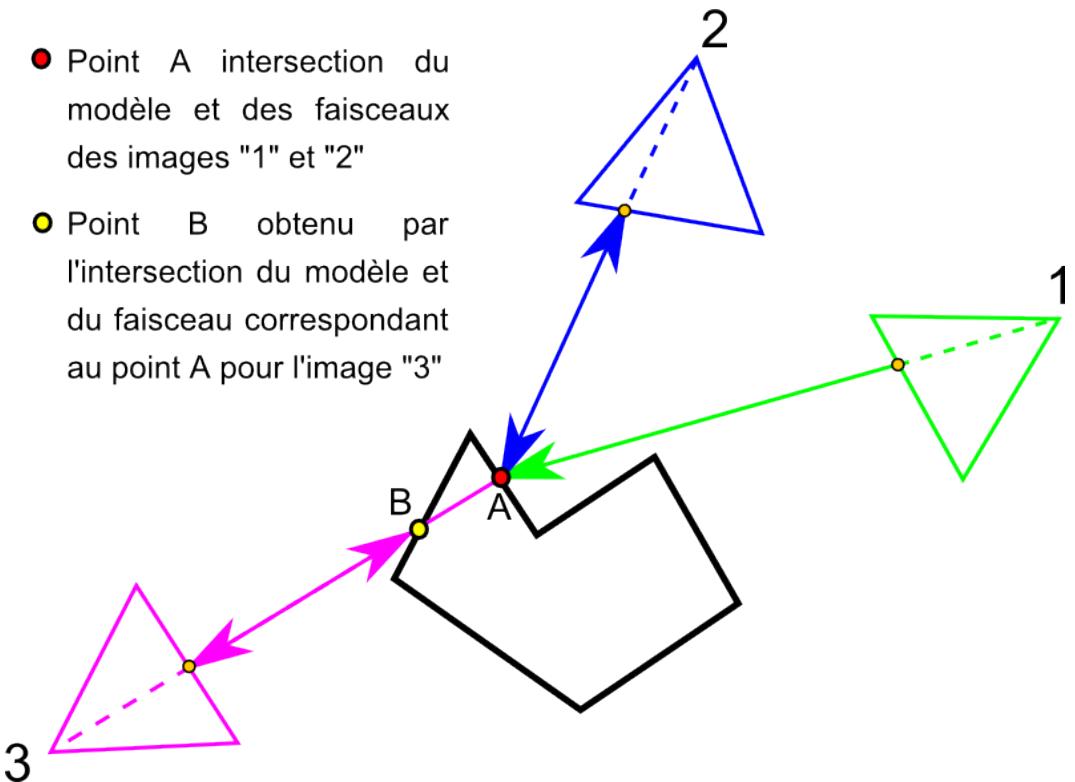


FIGURE 3.4 – Schéma de l'algorithme de détection des points de liaison

Par exemple, pour une image de 12Mpix sous résolue d'un facteur 16 comme celle présenté dans la figure 3.5, soit une dimension de 269\*179, on génère en général entre 200 et 1000 points de ce type, en fonction de la géométrie de la scène, du masque associé et des autres images considérées.



FIGURE 3.5 – 331 points de liaisons extrait de l'image 2 du jeu "Maison Miniature"

### 3.3.2 Limites du concept

Si cet algorithme de détection de points de liaison présente à la fois une bonne densité de points et une excellente robustesse, il présente également un problème handicapant : une forte dépendance à la qualité des nuages de points/cartes de profondeur sur lesquelles il s'appuie. En effet, la qualité de la collocation des différents sous-nuages desquelles les points sont extraits peut résulter en l'absence de détection de points de liaison.

D'autre part, dans certains rares cas, l'algorithme peut sélectionner des outliers si des nuages qui ne devraient pas se toucher se touchent effectivement. Cependant, dans ce cas, si la correction n'a plus aucun sens physique, la rupture radiométrique se verra tout de même réduite. Le risque étant une propagation de cette mauvaise correction.

## 3.4 Choix du modèle

### 3.4.1 Modèle multiplicatif

Une première tentative a été de rechercher un facteur multiplicatif constant à appliquer à chacun des canaux radiométriques de chaque image pour obtenir une illumination moyenne semblable dans tout le jeu d'images. Si cette approche permet une égalisation globale, corrigeant ainsi des différences de balance des blancs ou de niveau de luminosité, elle ne permet pas de modéliser des effets d'illumination non homogènes (de type flash par exemple, toujours centré dans la direction - variable - de prise de vue). Un modèle plus complexe et donc plus adaptable à des éclairages non homogènes doit donc être mis en œuvre.

### 3.4.2 Modèle polynômial

Le premier type de modèle complexe testé a été un polynôme de correction par image, fonction des coordonnées (X,Y) du point. Avec G le niveau de gris du canal étudié, on pose :

$$G_{cor}(X, Y) = G(X, Y) * \text{polynôme}(X, Y)$$

Le degré de ce polynôme peut varier en fonction de la complexité de la scène. Une suite de tests a amené à préférer un polynôme de degré 3. On résoudra donc par des techniques d'estimation linéaire un système d'équations constitué à partir des points de liaison radiométrique, chacun de ces points procurant une équation de type (avec  $(X_i, Y_i)$  la position du point et  $G_i$  son niveau de gris dans le canal considéré dans l'image i) :

$$G_1(X_1, Y_1) * \text{polynôme}_1(X_1, Y_1) = G_2(X_2, Y_2) * \text{polynôme}_2(X_2, Y_2)$$

**Problème lié au modèle polynômial** Ce type de modèle présente un défaut flagrant : le système a pour solution triviale le polynôme nul, la résolution par moindres carrés résulte donc systématiquement à l'obtention de cette solution. En effet, si tous les points sont noirs, les différences de radiométrie dans les zones étudiées sont effectivement nulles. Ce résultat n'est évidemment pas recevable, toute l'information radiométrique étant perdue au cours du processus.

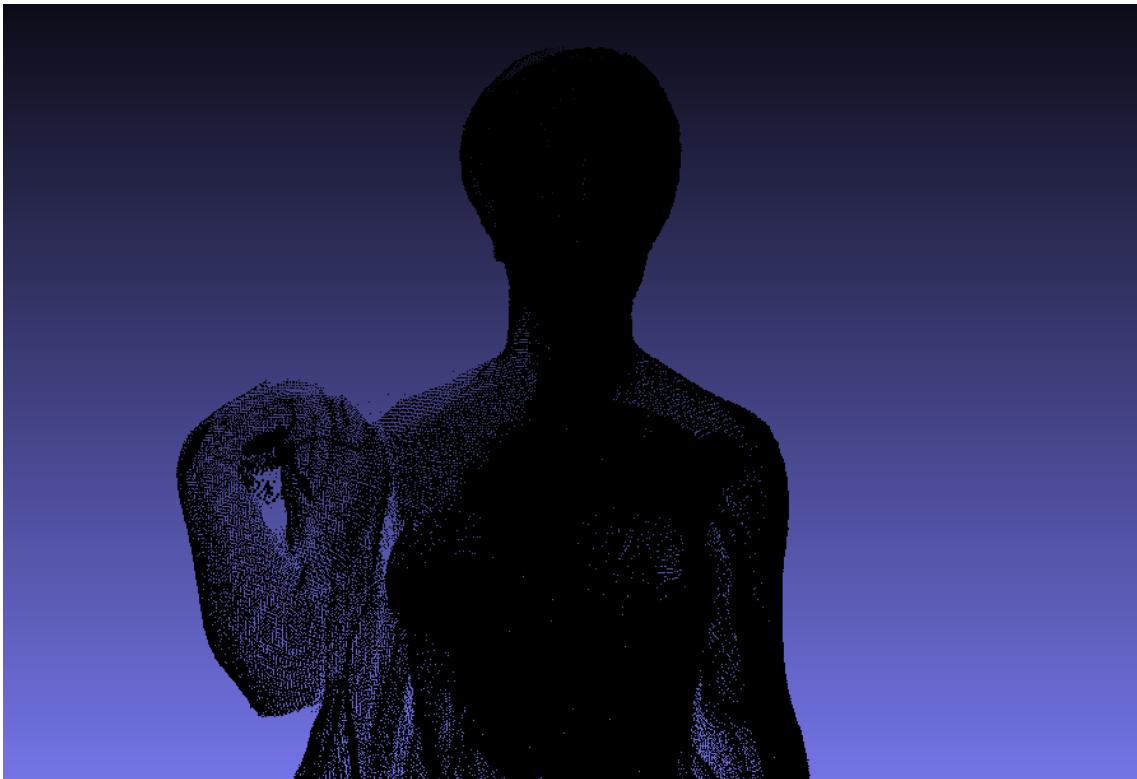


FIGURE 3.6 – Fusions de nuages égalisés "en noir"

Pour éviter la solution nulle, il convient alors d'ajouter dans le processus de moindres carrés des équations de poids fort contredisant cette possibilité (avec  $\alpha$  le coefficient de degré 0 des polynômes de correction) :

$$\alpha_i = 1$$

Cette solution n'est cependant pas suffisante, car un polynôme non nul brûlant (en mettant les pixels à la valeur maximale de 255) ou noircissant la zone d'intérêt est toujours possible. De nombreux tests utilisant les moindres carrés, l'algorithme de RANSAC normal ou une version modifiée affectant le score d'une solution en fonction du nombre de pixel brûlé/noirci ont montré l'inefficacité de la méthode.



FIGURE 3.7 – Images mal corrigées : image noircie (gauche) et brûlée (droite)

### 3.4.3 Une approche plus locale, le modèle interpolé

#### 3.4.3.1 Justification de l'approche

Les défauts que l'on cherche à corriger ont en fait des origines très locales : non lambertiennité des matériaux, variabilité de l'éclairage... Il semble donc intéressant d'explorer des méthodes de corrections plus sensibles aux voisinages qu'à la globalité de l'image.

#### 3.4.3.2 Algorithme

Pour chaque image, on se propose de déterminer une surface multiplicative non pas paramétrée mais interpolée sur les points de liaisons de type :

$$\left\{ \begin{array}{l} X \text{ et } Y \text{ les coordonnées en pixel du points dans l'image.} \\ K \text{ le rapport } G_j/G_i \text{ avec } i \text{ l'image traitée et } j \text{ l'autre image du couple ayant généré le point.} \end{array} \right.$$

La méthode d'extraction de points de liaison présentée en 3.3 produit extrêmement peu de points géométriquement aberrants, et il est peu probable que ces points présentent une aberrance colorimétrique marquée. Un tel point aberrant verrait de plus son effet en partie moyenné par les points adjacents. Dans le cadre de notre étude, il a donc été choisi de ne pas développer et mettre en place une procédure de filtrage.

On va ensuite générer une matrice de réels de même dimension que les images ayant servies à l'extraction des points de liaison. Pour tout  $(X, Y)$  de cette matrice, une valeur du facteur de correction est interpolée à partir des points de liaison décrit ci-dessus. L'interpolation se fait par pondération par l'inverse de la distance selon la formule suivante (où  $n$  est le nombre de points de liaison pour l'image concernée,  $(X_i, Y_i, K_i)$  les valeurs du point de liaison courant) :

$$Cor(X, Y) = \sum_{i=1}^n \frac{K_i}{\sqrt{(X_i - X)^2 + (Y_i - Y)^2}}$$

Une valeur multiplicative devant ensuite être appliquée à chaque point de l'image corrigée en sortie, elle est interpolée bilinéairement à partir de cette grille sous-échantillonnée.

#### 3.4.3.3 Fonctionnement pour plus d'une image

Si une image à plus d'une "voisine", sa correction va être influencée par toutes ses voisines. Les points de liaison étant réciproques, les influences sont croisées. Ainsi, si lors du traitement d'une image "A", un point de liaison triple est plus clair sur l'image "B" et plus sombre sur l'image "C", les influences vont être moyennées et le point peu affecté. Par contre, les images "B" et "C" se verront égalisées pour se rapprocher de la moyenne des trois images.

Les figures 3.8 et 3.9 montrent les surfaces de correction de deux images adjacentes, influencées par les autres images périphériques. La zone de gauche de l'image 3 n'est que très légèrement multipliée car l'image 4 est de couleur semblable, sauf sur le haut de la cheminée où l'image 2, plus claire, a une influence. Sur la droite de l'image, le facteur de multiplication est plus fort (environ x1.35) pour se mettre en accord avec l'image 2 qui se voit parallèlement fortement divisée (x0.75) sur cette même zone, l'écart moyen (le "K") de 2 vers 3 étant d'environ  $1.35/0.75 = 1.8$  sur cette zone.

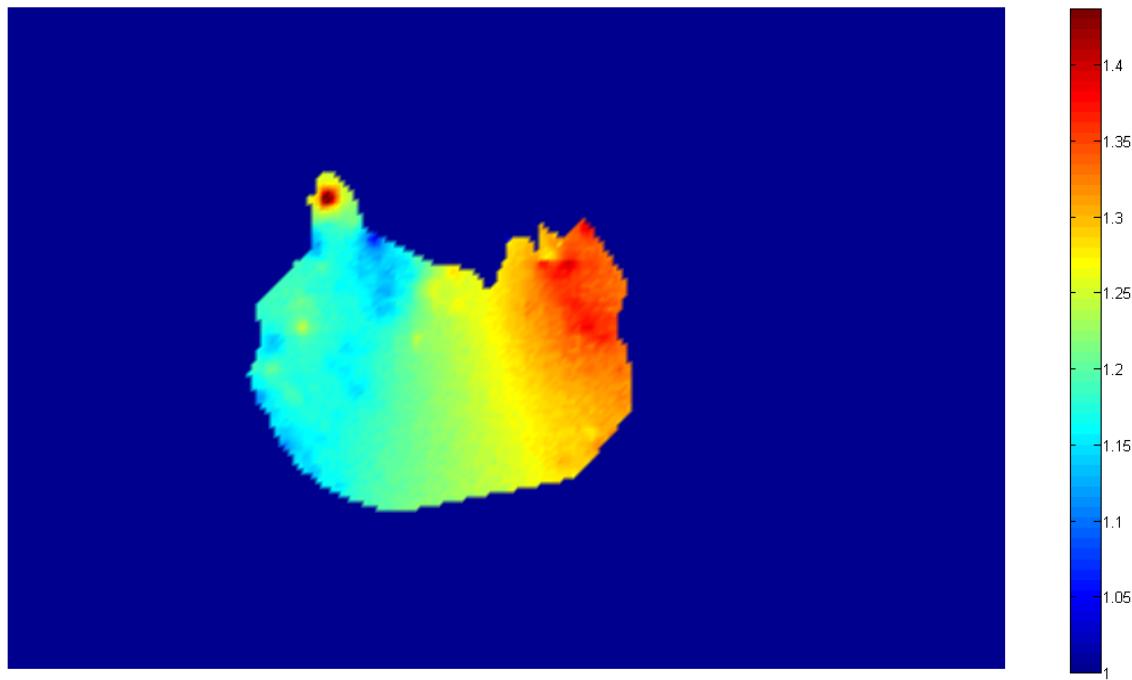


FIGURE 3.8 – Surface de correction pour l'image 3 du jeu "Maison Miniature"

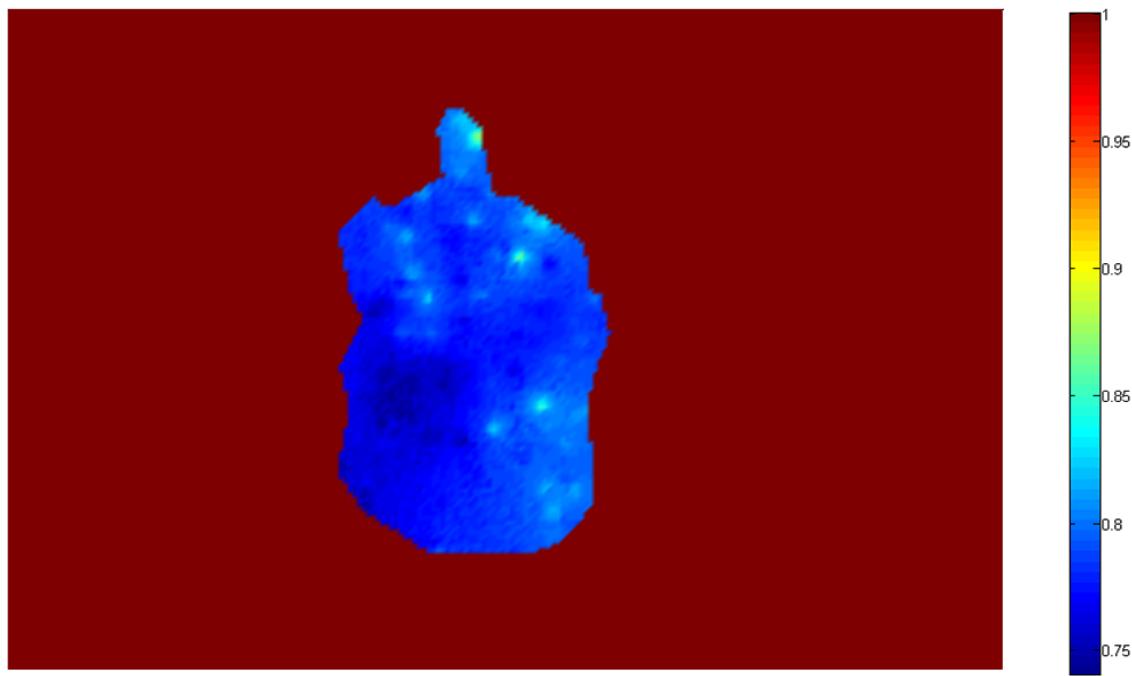


FIGURE 3.9 – Surface de correction pour l'image 2 du jeu "Maison Miniature"

### 3.5 Analyse des résultats

Bien que l'objectif des traitements ici présentés soit de portée principalement esthétique et que qualifier leur réussite se fasse surtout par la visualisation des résultats, une analyse statistique montre également les améliorations de cohérence apportées par les algorithmes.

La variable étudiée est la répartition des valeurs  $G_j/G_i$ , qui représentent la disparité entre les radiométries des images dont sont issus les points de liaison. Plus une valeur est proche de 1, plus la cohérence radiométrique est forte, et inversement.

Pour l'analyse, on va s'appuyer sur l'effet des algorithmes sur la minimisation d'un critère qu'ils cherchent explicitement à minimiser. Cependant, l'analyse tire sa valeur de l'étude de l'efficacité de cette minimisation.

#### 3.5.1 Présentation des jeux test

Pour tester les algorithmes, deux jeux test ont été exploités. Ils présentent des cas classiques pour lesquelles les corrections ici présentées sont nécessaires.

##### 3.5.1.1 Jeu "Baigneuse Maillol"

Le jeu d'image "Baigneuse Maillol" a été photographié par Adrien Didierjean (RMN) et initialement traité par Nicolas Martin-Beaumont (MAP GAMSAU). Il représente *La Baigneuse a la draperie* du sculpteur Aristide Maillol, d'après la modèle Dina Vierny, exposée dans le jardin des Tuilleries à Paris.



FIGURE 3.10 – 5 images principales du jeu "Baigneuse Maillol" - ©MAP

Il se compose de 25 images destinées à l'orientation des images et de 25 autres images destinées à la corrélation dense. Ce deuxième groupe d'images est subdivisé en 5 groupes de 5 images dont les images centrales sont présentées dans la figure 3.10. La statue occupe uniquement le deuxième tiers horizontale des images et l'arrière fond est relativement lointain.

L'éclairage des images a été qualifié par les fournisseurs des images comme *soumis à des nuages par intermittence*. De plus, les photos étant prises en extérieur, on remarque un phénomène de contre-jour. De ce fait, d'importantes variations de luminosité sont observées, les images étant très sombres sur le début de la prise de vue (Première ligne de la figure 3.10) et beaucoup plus claires après.

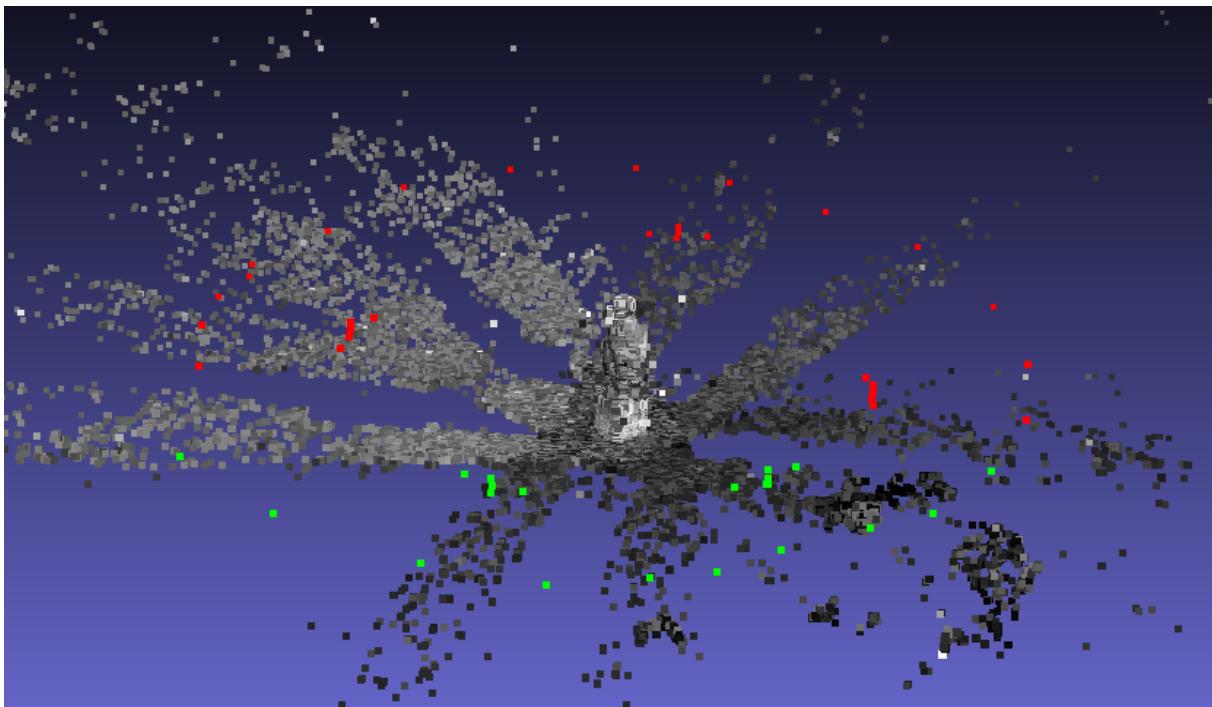


FIGURE 3.11 – "AperiCloud" du jeu "Baigneuse Maillol" - ©MAP

La prise de vues a été effectuée avec un Nikon D3X. Les paramètres choisis pour cette prise de vue étaient une priorité ouverture fixée à F/8, une focale de 50mm et un ISO-100.

### 3.5.1.2 Jeu "Maison Miniature"



FIGURE 3.12 – 5 images principales du jeu "Maison Miniature" - ©Luc Girod

Le jeu d'images "Maison Miniature" a été photographié et traité directement au laboratoire. Il représente une maquette de maison écossaise nommée "Fisherman's Wharf" de dimension approximative de 12\*7\*8cm, réalisée d'après un design de David Winter de 1983. L'objet a été prêté par Olivier Dissard qu'il convient de remercier ici.

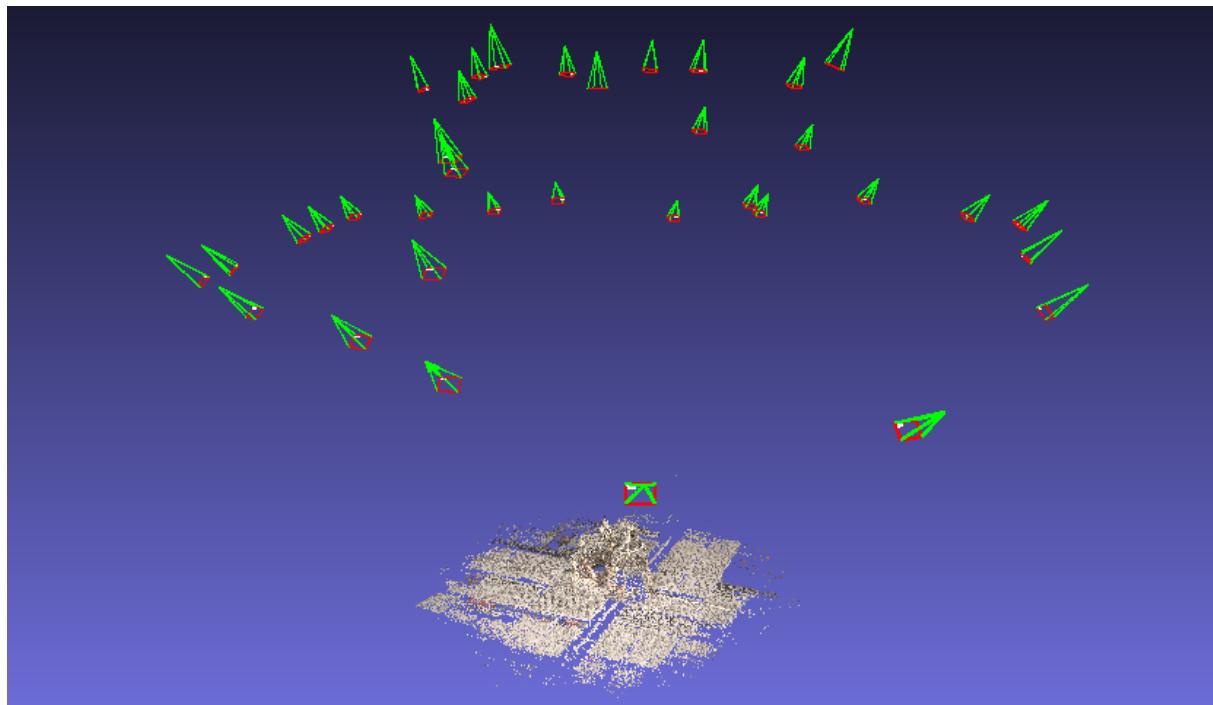


FIGURE 3.13 – "AperiCloud" du jeu "Maison Miniature" - ©Luc Girod

Il se compose de 36 images convergentes réparties en demi-sphère autour de l'objet. Les 5 groupes d'images permettant la corrélation dense ont été issus de cette demi-sphère et sont respectivement des vues de la face, de l'envers, des deux côtés et du dessus de l'objet. La maquette est centrée dans les images et occupe un peu plus d'un neuvième de celles-ci.

L'éclairage des images a légèrement subit l'ombre portée du photographe pendant la prise de vues pour les images "de face". La luminosité était cependant plutôt stable, aucun flash n'ayant été utilisé.

La prise de vues a été effectuée avec un Nikon D90. Les paramètres choisis pour cette prise de vues étaient une priorité ouverture fixée à F/5, une focale de 52mm et un ISO-200. Cette combinaison provoque un vignettage léger mais présent qui nécessite une correction, comme on le verra en [3.5.3.3](#).

### 3.5.2 Résultats du processus d'égalisation

Dans cette partie, on va exposer les résultats des traitements sur les deux jeux tests "Maison Miniature" (illustrés en 3.14) et "Baigneuse Maillol" (illustrés en 3.15) présentés en 3.5.1. Dans un premier temps ,on verra les résultats obtenus avec les paramètres par défaut, puis on justifiera ces choix par une comparaison des résultats obtenus avec des paramètres différents.

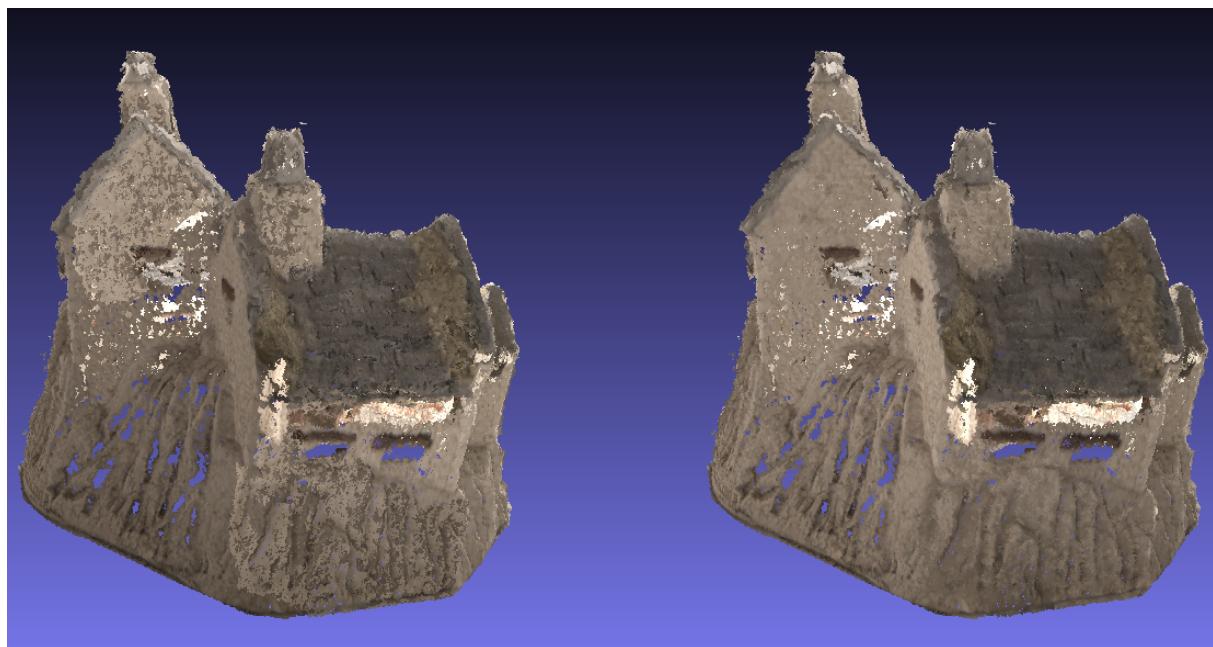


FIGURE 3.14 – Nuage original (gauche) et corrigé (droite) - Maison Miniature

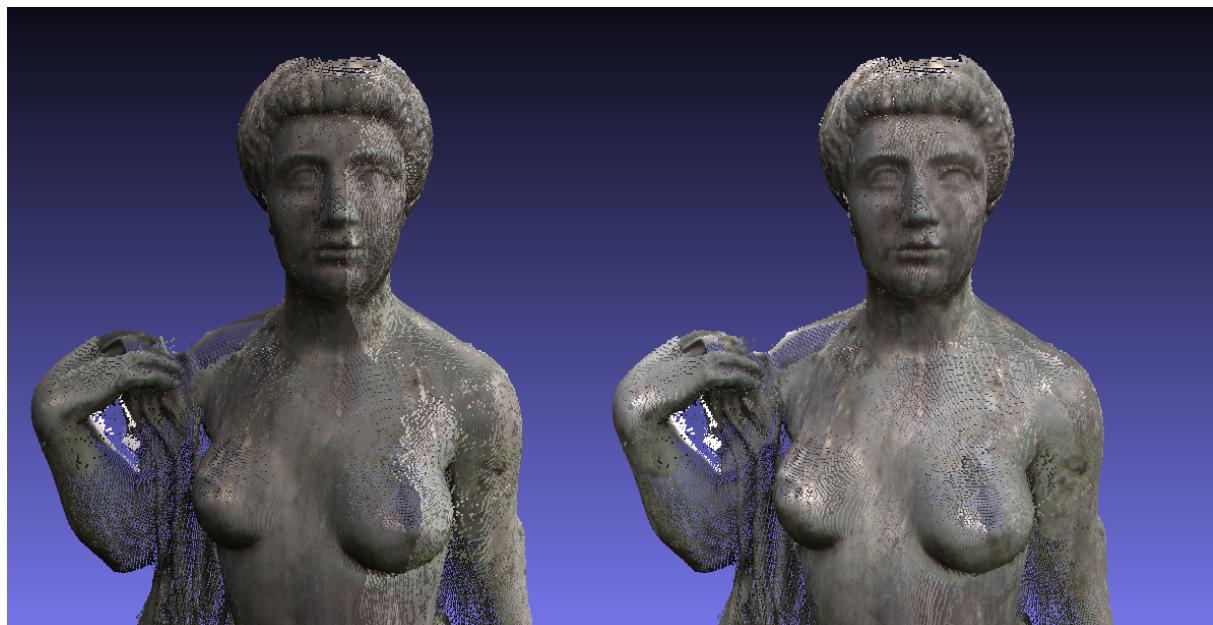


FIGURE 3.15 – Nuage original (gauche) et corrigé (droite) - Baigneuse Maillol

$G_j/G_i$	Original	Traité
Min	0,40	0,52
Max	2,25	1,81
Moyenne	1,03	1,01
Écart-type	0,23	0,13

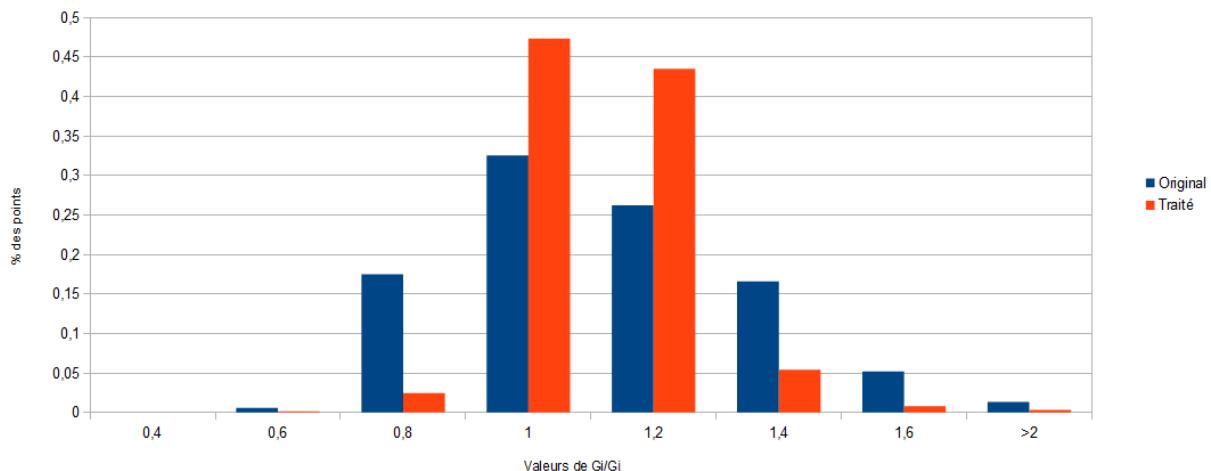
TABLE 3.1 – Statistiques du jeu "Maison Miniature"

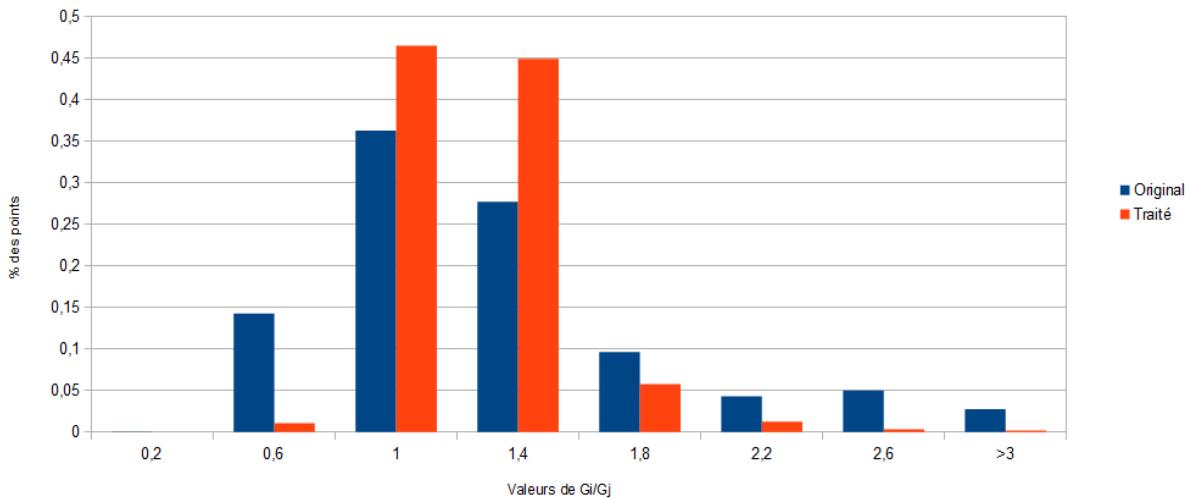
$G_j/G_i$	Original	Traité
Min	0,13	0,25
Max	8,09	3,14
Moyenne	1,13	1,05
Écart-type	0,66	0,26

TABLE 3.2 – Statistiques du jeu "Baigneuse Maillol"

Dans le tableau 3.2, on remarque que le traitement a considérablement réduit les écarts radiométriques entre points homologues, avec un écart-type divisé par 3. La présence de points aberrants est cependant indéniable. Le tableau 3.1 présente des améliorations également importantes et un résultat véritablement satisfaisant comme on peut le voir dans la figure 3.14. Il est cependant important de noter que l'état initial était moins inégal.

Les figures 3.16 et 3.17 montrent que la répartition des facteurs  $G_j/G_i$ , bien plus centrée après traitement. Cette concentration des valeurs autour de 1 indique qu'une large majorité des points de liaisons radiométriques ont une couleur très proches dans les différentes images dont ils sont issus, ce qui est l'objectif des traitements.

FIGURE 3.16 – Répartition des 1868 facteurs  $G_j/G_i$  - Maison Miniature

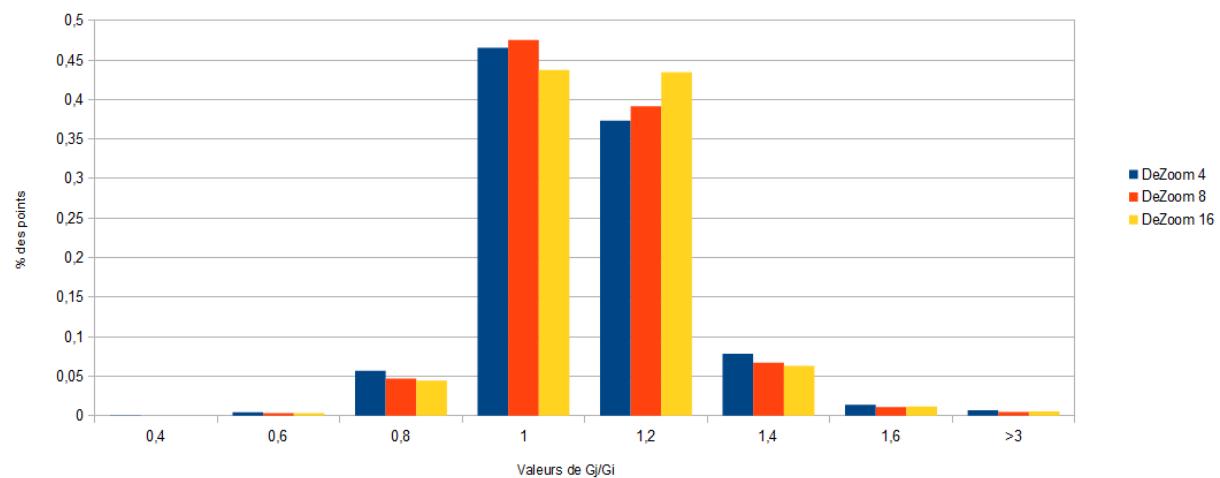
FIGURE 3.17 – Répartition des 3200 facteurs  $G_j/G_i$  - Baigneuse de Maillot

### 3.5.3 Influences des paramètres des algorithmes

Les algorithmes utilisés présentent un certains nombre de paramètres réglables. Le choix des paramètres par défaut va être ici présenté. Si la qualité du résultat a toujours été la priorité, la variation du temps de calcul résultant de la modification d'un paramètre a également été prise en compte.

#### 3.5.3.1 Influence de la résolution

Il a été écrit lors de la présentation des algorithmes en 3.3 que la recherche de points de liaison produisait des données suffisantes dès une sous-résolution de facteur 16. Le tableau 3.3 et la figure 3.18 montrent que la qualité du traitement des images n'est pas meilleure pour une résolution plus grande. En effet, l'écart type ne se réduit pas lorsque l'on passe d'une sous-résolution d'un facteur 16 à un de facteur 4, il augmente même légèrement dans cet exemple.

FIGURE 3.18 – Répartition des facteurs  $G_j/G_i$  en fonction de la résolution - Maison Miniature

De plus, quand le DeZoom 16 permet de corriger une image en moins de 2 secondes, il faut plus de 6 minutes pour le DeZoom 4 (et plusieurs heures en DeZoom 2). Ceci s'explique par un

algorithme avec une complexité en  $O(n^4)$  car le nombre de points détectés est environ trois fois plus grand à chaque doublement de résolution et que la dimension de la surface sur laquelle les données sont interpolées est parallèlement multipliée par 4.

$G_j/G_i$	DeZoom 4	DeZoom 8	DeZoom 16
Min	0,31	0,44	0,44
Max	3,84	4,63	2,12
Moyenne	1,01	1,01	1,02
Écart-type	0,17	0,16	0,15
Nb Points	17979	6464	1868

TABLE 3.3 – Statistiques du jeu "Maison Miniature" en fonction de la résolution

### 3.5.3.2 Influence de la définition de proximité

Le tableau 3.4 montre l'influence d'une modification de la notion de proximité lors du filtrage des points de liaison. En abaissant ce critère, on accepte la présence de points dont la fiabilité est moindre. On obtient environ deux fois plus de points en multipliant le critère par deux. Cependant, ce surplus de points n'améliore en rien les résultats et introduit plus d'outliers (que l'on remarque par l'évolution des minimaux et maximaux).

$G_j/G_i$	1/4	1/8	1/16
Min	0,22	0,23	0,44
Max	5,06	4,86	4,63
Moyenne	1,01	1,01	1,01
Écart-type	0,16	0,16	0,16
Nb Point	25756	12978	6464

TABLE 3.4 – Statistiques du jeu "Maison Miniature" en fonction de la définition de la proximité

La figure 3.19 montre la répartition des facteurs  $G_j/G_i$  mettant très légèrement en avant le choix de la distance maximum valable à  $\frac{1}{16}$  de la résolution d'un pixel. Cet avantage n'est pas flagrant mais atteste de la suffisance du nombre de points généré avec cette valeur. Bien sûr, le temps de calcul est aussi affecté par ce paramètre : le temps de calcul de la matrice de correction étant linéaire par rapport au nombre de points de liaison radiométriques.

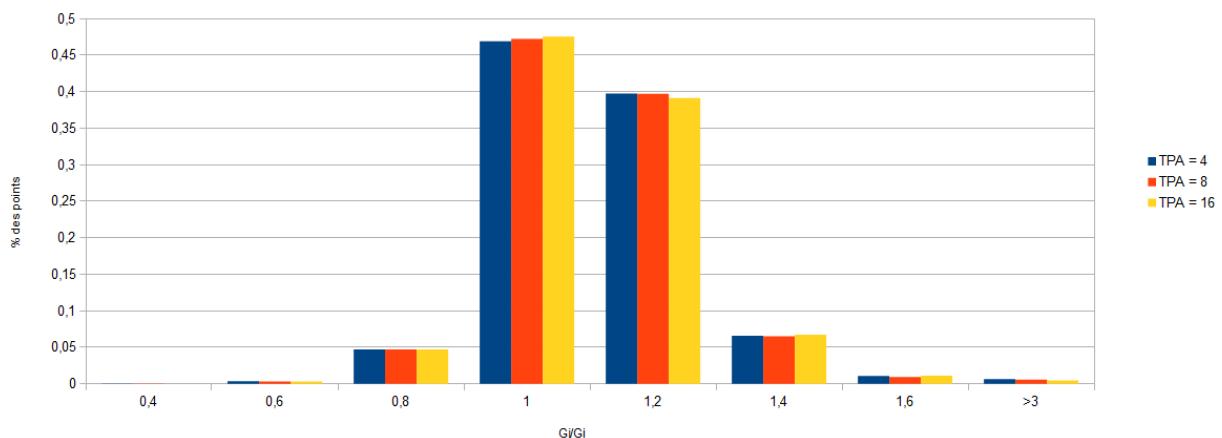


FIGURE 3.19 – Répartition des facteurs  $G_j/G_i$  en fonction de la proximité - Maison Miniature

### 3.5.3.3 Influence de la correction du vignettage

Comme vu en 2, la vignettage des images est un aspect important de la correction radio-métrique. Cependant, au vu de l'approche locale des correction, il pourrait sembler superflu d'effectuer cette correction. Cependant, le vignettage apparaît être un facteur perturbateur et les résultats obtenu en tenant compte de sa présence sont légèrement meilleurs (voir le tableau 3.5 et la figure 3.20). On remarque que les statistiques d'entrée ne sont qu'anecdotiquement différentes, la perturbation n'étant que légère sur la zone d'intérêt pour ce jeu de données, mais suffisante pour influencer la qualité de la correction.

$G_j/G_i$	Original Avec	Original Sans	Traité Avec	Traité Sans
Min	0,40	0,40	0,52	0,43
Max	2,25	2,27	1,81	2,20
Moyenne	1,03	1,03	1,01	1,02
Écart-type	0,23	0,23	0,13	0,15

TABLE 3.5 – Statistiques du jeu "Maison Miniature" avec et sans correction du vignettage

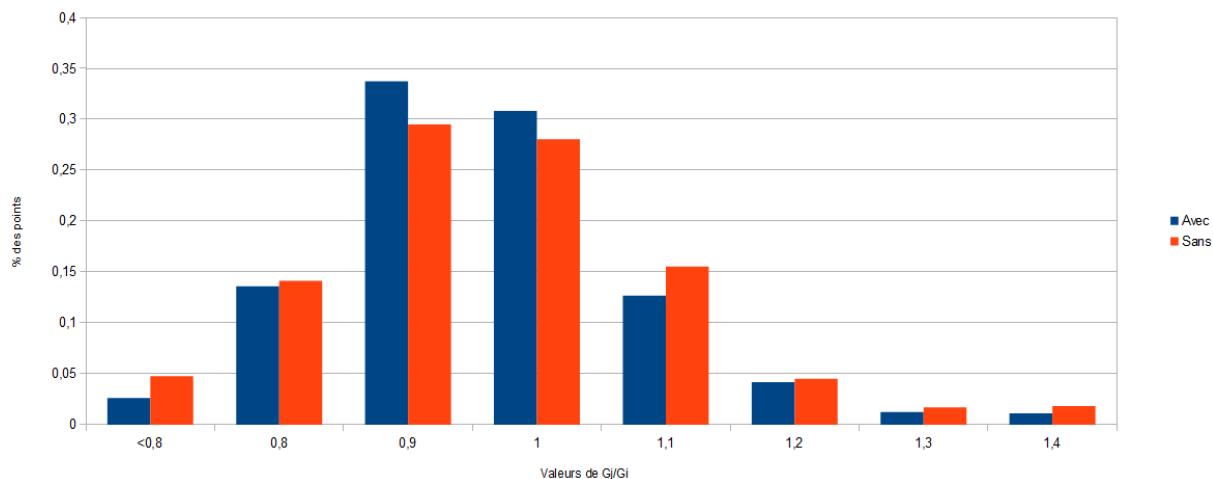


FIGURE 3.20 – Répartition des facteurs  $G_j/G_i$  avec et sans dévignettage - Maison Miniature

## 3.6 Conclusion : améliorations possibles et pistes de recherche

Nous avons vu dans ce chapitre des algorithmes d'égalisation radiométriques d'ensembles d'images dans le cadre de la fusion de nuages de points 3D. Ceux-ci présentent des résultats convaincants tant visuellement que statistiquement. Il est cependant bien sûr possible d'envisager des concepts plus précis, efficaces et polyvalents, que la durée du stage n'a pas permis d'aborder.

### 3.6.1 Estimation du BRDF

Un premier concept qui me semble intéressant d'explorer est une correction par estimation du BRDF (Bidirectional Reflectance Distribution Function, cf figure 3.21). En effet, la correction alors effectuée serait basée sur la physique des objets et seraient donc plus propres, faisant fi de l'éclairage lors de la prise de vue. Cependant, une telle correction nécessite une estimation d'un paramètre complexe pour chaque point terrain, et donc des données en entrée très volumineuses qui ne sont pas produites par MicMac dans le cadre de la chaîne "normale".

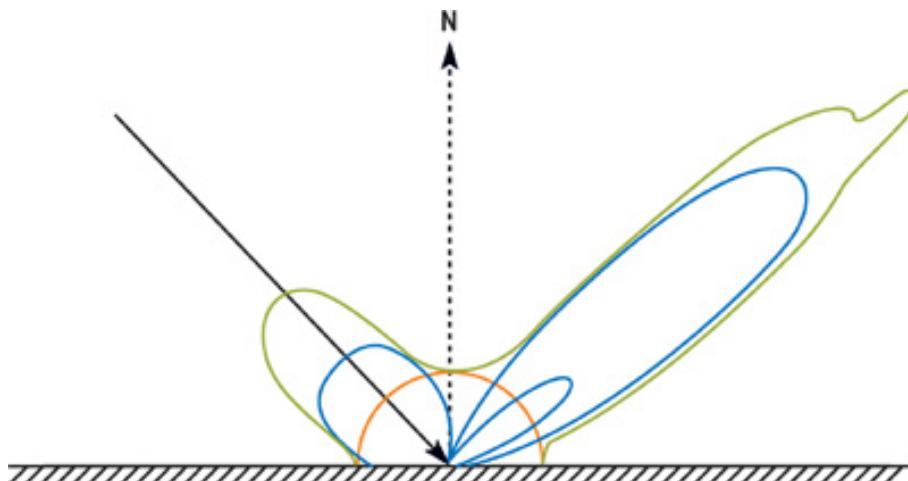


FIGURE 3.21 – Différents BRDF d'un point ©David McAllister, NVIDIA

### 3.6.2 Blending 3D

Un autre concept qui me semble prometteur est le blending 3D. Il consisterait à repérer les points proches et ajuster/moyenner leurs radiométries. L'approche simple ne serait cependant pas suffisante, l'intersection d'un nuage blanc et d'un nuage noir donnerait un nuage gris et l'impression de trois zones distinctes. La problématique serait donc de réussir à mettre en place un système de moyennes pondérées en fonction de la proximité des points aux nuages adjacents, par exemple en fonction de la distance du point au centre de gravité de chaque sous nuage.

# IMPLÉMENTATION DES ALGORITHMES

CHAPITRE  
**4**

Ce stage s'inscrivant dans le développement du logiciel MicMac, les algorithmes décrits ci-dessus ont été implémentés dans la librairie MicMac sous la forme de nouvelles commandes.

## 4.1 Prise en main des méthodes de développement

### 4.1.1 Tests sous Octave

Dans un premier temps les algorithmes de correction du vignettage ont été implémentés et testés sous Octave. Cependant, la gestion des images n'est pas aisée dans ce langage, nécessitant une pré-conversion en `.mat`. De plus, les calculs relativement lourds nécessaires à l'exécution de ces algorithmes et le volume de données à traiter se sont révélés être incompatible avec le langage, beaucoup trop lent et coûteux en mémoire.

Il a donc rapidement été choisi de travailler directement en C++, le rendu final du projet étant de toutes façons attendu dans ce langage.

### 4.1.2 La librairie MicMac

La librairie MicMac est une librairie photogrammétrique en C++ initialement développée par Marc Pierrot-Deseilligny. Elle se compose de plus de 800 fichiers sources et il serait irréaliste de vouloir la maîtriser dans ses moindres détails. Cependant, l'un des objectifs du stage ici présenté étant la création de nouvelles fonctions dans cette librairie, une connaissance élémentaire de son fonctionnement est nécessaire.

La librairie est extrêmement riche, elle contient en effet des systèmes de lecture et d'écriture d'images, de manipulation de points de liaison, d'estimation linéaires...

Ayant déjà développé les fonctions Drunk et Apero2PMVS lors de projets précédents, le fonctionnement général m'était déjà familier. Cependant, de nombreuses fonctions dont les commandes présentées en [4.2](#) et [4.3](#) font usage ne m'étaient pas connues. Je citerais comme exemple le moteur de résolution par moindres carrés ou les fonctions de lectures des points de liaison. Un approfondissement de mes connaissances de la librairie a donc été ponctuellement nécessaire.

### 4.1.3 Développement coopératif avec Mercurial

La librairie MicMac est open source et son développement est collaboratif. Sont actuellement cités par la bannière de fin d'exécution du programme comme développeurs : Marc Pierrot-Deseilligny, Matthieu Deveau, Jérémie Belvaux, Gerald Choqueux et Luc Girod (voir figure [1.1](#)).

Pour permettre une telle collaboration, ainsi qu'une mise à disposition des sources en direct aux utilisateurs de ce projet open source, un dépôt Mercurial est en place à l'adresse <https://geoportail forge.ign.fr/hg/culture3d>.

Au cours du développement, une publication sur le dépôt, servant à la fois de système de diffusion et de sauvegarde, est effectuée régulièrement : tous les soirs si la version actuelle "compile" et à chaque changement important. Depuis le début du stage, MicMac est passé de la révision 950 à plus de 1600.

## 4.2 V.O.D.K.A.

### 4.2.1 Nom et fonction

La commande de correction du vignettage a hérité d'un nom d'alcool conformément à la thématique utilisée pour le reste de la librairie : Vodka. La décomposition de l'acronyme (qui s'affiche en fin d'exécution de la commande) est *Vignette Of Digital Kamera Analysis*. Elle implémente les algorithmes décrit au chapitre 2.

### 4.2.2 Données en entrée

L'utilisation de cette commande nécessite que le calcul des points de liaison ait déjà été effectué (via la commande Tapioca), les algorithmes s'appuyant sur ces points. L'appel de la commande se fait simplement par la ligne suivante (où *ImagesPattern* est l'expression régulière décrivant la liste d'images à exploiter) :

```
mm3d Vodka ImagesPattern
```

### 4.2.3 Gestion des jeux d'images

Le vignettage étant un effet dépendant de la focale et de l'ouverture utilisés par la caméra lors de la prise de vue, il convient de découper un jeu d'images éventuellement non homogène en sous-jeux d'images partageant les mêmes paramètres. Cela se fait par l'analyse des exifs. Les sous-groupes sont alors traités indépendamment, comme si la commande avait été appelée avec ces images uniquement.

D'autre part, l'ISO et le temps d'exposition de chaque image, affectant la luminosité des images, sont enregistrés et les plus grandes valeurs sont extraites pour être utilisées comme maximums dans la correction décrite en 2.4.3.

La présence d'une calibration pour un couple Focale/Ouverture est ensuite vérifiée si l'utilisateur a spécifié un dossier de calibration. Si une telle calibration existe, elle n'est pas recalculée.

### 4.2.4 Données en sortie

Par défaut, pour chaque sous-groupe d'images, la commande génère un fichier tif de flottants des coefficients multiplicateurs de la taille des images en entrée nommé *Vignette/Foc0000Dia111.tif*, où 0000 est la focale en millimètres (en quatre chiffres) et 111 l'ouverture multipliée par 10 (en trois chiffres).

Des images corrigées sont générées seulement sur demande de l'utilisateur, la calibration pouvant être faite avec des images ne nécessitant pas d'exploitation future ou par un sous ensemble des images, une correction globale ayant alors lieu plus tard dans la chaîne de traitement.

### 4.2.5 Options

La commande propose plusieurs options :

- *DoCor* (*bool*) qui active la correction des images
- *InCal* (*string*) qui définit le dossier contenant le ou les fichiers tif issus d'une précédente calibration
- *Out* (*string*) qui définit le répertoire de sortie

### 4.2.6 Temps de traitement

Pour estimer les temps de calcul, on utilisera le jeu d'images "Maison Miniature" (décris en 3.5.1.2). Il est important de noter que le développement des images (conversion en .tif monochromes ou colorés) n'est pas inclus dans ces estimations.

Le temps de calcul peut se diviser en plusieurs étapes :

- *Lecture des metadonnées EXIF* : 14 secondes
- *Lecture des points de liaison* : 120 secondes
- *Estimation de la vignette* : 27 secondes
- *Correction des images* : 45 secondes

## 4.3 A.R.S.E.N.I.C

### 4.3.1 Nom et fonction

La commande d'égalisation radiométrique se nomme du nom d'un poison plus dangereux que l'alcool : Arsenic. La décomposition de l'acronyme (qui s'affiche en fin d'exécution de la commande) est *Automated Radiometric Shift Equalization and Normalization for Inter-image Correction*. Elle implémente les algorithmes décrit au chapitre 3.

### 4.3.2 Données en entrée

Pour permettre la génération des points de liaison décrite en 3.3, chacune des images qui sera égalisée doit être associée à une carte de profondeur, générée par la commande MicMac/Malt. Les images concernées seront identifiées par une expression régulière et la commande sera appelée de la façon suivante :

*mm3d Arsenic ImagesPattern*

Si l'option est choisie, la commande va appeler la commande Vodka et utiliser les images corrigées plutôt que les images initiales pour tout le traitement. L'appel effectué est (avec InVig un dossier défini dans les options de la commande) :

*mm3d Vodka ImagesPattern DoCor = 1 Out = InVig InCal = InVig*

### 4.3.3 Données en sortie

En sortie, la commande écrit les images corrigées dans un dossier (par défaut "Egal/") qui peuvent être utilisées pour la coloration d'un nuage de point via la commande Nuage2Ply.

### 4.3.4 Options

La commande propose plusieurs options :

- *TPA (Tie Point Accuracy - int)* qui définit la précision du filtre sur les points de liaison (par défaut à 16, pour  $\frac{1}{16}$  de la résolution du pixel)
- *ResolModel (int)* qui définit la résolution du modèle utilisé pour le calcul des points de liaison (par défaut à 16 pour DeZoom 16)
- *InVig (string)* qui définit le dossier de calcul des vignettes
- *Out (string)* qui définit le répertoire de sortie

### 4.3.5 Temps de traitement

Pour estimer les temps de calcul, on utilisera les 5 images centrales du jeu d'images "Maison Miniature" (décrit en 3.5.1.2) avec les paramètres par défaut. Il est important de noter que le développement des images (conversion en .tif monochromes ou colorés) n'est pas inclus dans ces estimations.

Le temps de calcul peut se diviser en plusieurs étapes :

- *Création des images sous-résolues* : 1 secondes
- *Extraction des points de liaison* : 3 secondes
- *Correction des images* : 9 secondes

## 4.4 Parallélisation

Une partie des traitements effectués ne présente pas d'interdépendance entre objets, la partie de correction des images dans Arsenic par exemple. Il n'y a donc aucun problème empêchant la parallélisation de ces processus.

Pour ce faire, on utilise la librairie OpenMP, qui permet très aisément de paralléliser des boucles FOR par la simple adjonction de la ligne *#pragma omp parallel for* avant la boucle. On gagne ainsi un facteur égal au nombre de processeur disponibles sur le temps de calcul pour cette partie du traitement.

## 4.5 Conclusion

Les algorithmes ont donc été implémentés avec succès dans la librairie MicMac, cependant, leur véritable intégration dans la chaîne de traitement est encore à l'étude. Les commandes pourraient être appelées sous formes d'option par les autres commandes, un appel de Vodka par Tawny semblerait par exemple une possibilité.

# Conclusion

---

A l'issu des quatre mois et demi de stage, on peut affirmer que les objectifs ont été atteints. En effet, les algorithmes permettant l'égalisation radiométrique de jeux d'images pour permettre une bonne coloration de nuages de points et des transitions entre nuages adjacents ont été développés avec succès. De plus, ils ont été implémentés en C++ et intégrés à la librairie MicMac.

Cependant, les algorithmes ici présentés ne sont pas parfaits et peuvent être mis en échec dans certains cas, notamment sur des objets trop brillants présentant des réflexions spéculaires ou de variations radiométriques trop importantes. De plus, si l'implémentation des algorithmes est effective, ils ne sont pas encore véritablement intégrés à la chaîne de traitement MicMac. Leurs sorties n'étant en effet prises en compte par les autres commandes ni automatiquement ni via une option.

Si ce stage constitue une étape dans la recherche de méthodes d'égalisation radiométrique de nuages de points, il semble clair que ce domaine nécessite encore l'attention des photogramètres. Attention qu'il me semble mériter, la popularité des produits de type nuage de points et leurs maillages associés étant grandissante. Par exemple, la reproduction par impression 3D nécessite une bonne qualité radiométrique. Je ne peux donc qu'espérer que l'occasion d'approfondir ce domaine et d'explorer de nouvelles pistes se présentera au plus vite mettant à la disposition de la communauté des outils et algorithmes encore plus efficaces et polyvalents.

Pour simplifier le travail de mes successeurs, j'irai jusqu'à proposer un nouveau nom pour une commande future plus efficace que mon ARSENIC présenté dans ce mémoire :

**CIGUE** - *Color of ImaGes Utopic Equalisation*



# Bibliographie

---

- [1] Laure Chadelier and Gilles Martinoty. A radiometric aerial triangulation for the equalization of digital aerial images and orthoimages. *Photogrammetric Engineering and Remote Sensing Vol. 75, No. 2, pp 193-200*, Feb 2009.
- [2] Martin A. Fischler and Robert C. Bolles. Random sample consensus : A paradigm for model fitting with applications to image analysis and automated cartography. *Comm. Of the ACM, vol. 24, pp. 381-395*, Jun 1981.
- [3] Dan B Goldman and Jiun-Hung Chen. Vignette and exposure calibration and compensation. *ICCV'05*, Oct 2005.
- [4] David Lowe. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision, 60, 2, pp. 91-110*, 2004.
- [5] Marc Pierrot-Deseilligny. *MicMac, Apero, Pastis and Other Beverages in a Nutshell!* LGA, IGN, Champs-Sur-Marne, France, In Progress - latest version Jun 2013.



# **Annexes**

A Documentation et code source publié . . . . .	51
---	----



# DOCUMENTATION ET CODE SOURCE PUBLIÉ

---

ANNEXE

**A**

Le code C++ développé lors du stage ainsi que la documentation utilisateur officielle sont disponibles sur le dépôt Mercurial du projet Culture 3D :

<https://geoportail.forge.ign.fr/hg/culture3d>

Les fichiers sources concernés sont dans le dossier src/uti\_image et sont les suivants :

- *Arsenic.h* : qui défini les objets utilisés
- *CPP\_Arsenic.cpp* : qui contient le script et les fonctions liés à la commande A.R.S.E.N.I.C.
- *CPP\_Vignette.cpp* : qui contient le script et les fonctions liés à la commande V.O.D.K.A.