

Rapport CST : *Détection de bruit dans les
images de synthèses stéréoscopiques par
méthodes d'apprentissage automatiques*

Jérôme BUISINE

July 2019

Table des matières

1	Introduction	9
1.1	Présentation de la thématique	10
1.2	Contexte de la thèse	10
1.3	Problématique abordée	10
2	Bibliography	13
2.1	Bibliographie	14
3	Travaux réalisés	15
3.1	La base de données d'images	16
3.2	L'approche décomposition SVD	19
3.2.1	Decription de la méthode	19
3.2.2	Sélection des attributs	19
3.2.3	Etude des échantillons de rendu	21
3.2.4	Les approches futures	22
4	Enseignements	23
5	Formation	25
5.1	Conclusion	25
	Appendices	27
A	Transformation L^*a^*b	29
B	Autre annexe	31

Table des figures

1.1	Intéraction du modèle de détection de bruit lors du rendu d'image	11
3.1	Découpage de l'image d'une scène pour traitement	16
3.2	Extraction de seuil perceptif pour une zone de la scène	17
3.3	Processus de génération des bases d'apprentissage, de validation et de test	18
3.4	Décomposition en valeurs singulières	19
3.5	Aperçu des valeurs singulières sur la scène <code>Appart1opt02</code>	20

Liste des tableaux

Chapitre 1

Introduction

1.1 Présentation de la thématique

Les moteurs de rendus d'illuminations globales sont utilisées de nos jours dans le but de générer des images photo-réalistes. La représentation d'une scène modélisée en 3D fournie en entrée au moteur de rendu permet l'obtention d'une image en sortie de celui-ci. Il définit la valeur de chaque pixel de l'image suivant les propriétés physiques des objets qui constituent la scène. Plusieurs algorithmes de rendus ont été développés durant ces dernières années dans le but d'approcher l'image finale souhaitée. Chaque méthode possède ses propres avantages et inconvénients, parfois coûteuse en temps ou pas assez précise si le temps est en manquant. Ces méthodes de rendus se basent toutes sur une méthode d'approximation Monte-Carlo (suivant la loi des grands nombres) dans le but d'approcher l'équation de rendu [?] et chaque valeur de pixel de l'image par le biais d'une moyenne empirique de l'approximation des valeurs des pixels durant le rendu.

Dans l'objectif d'évaluer la qualité d'une image, des approches et métriques de calculs ont été développées et proposées dans la littérature. Ces métriques sont parfois spécifiques pour des distortions d'images telles que le flou gaussien, JPEG2000, compression JPEG bruitée, ringing. Généralement ciblé sur du bruit additif. Globalement, les approches publiées sont appliquées pour la mesure de qualité d'une image dites naturelle et très peu pour mesurer la qualité d'une image photo-réaliste. En effet, lors du rendu de l'image, un bruit peut être perceptible par l'humain. Il ne s'agit ici en aucun cas d'un bruit additif mais plutôt un bruit dit « impulsif », dû au fait de la complexité d'approximation des valeurs finales de pixels dans une zone de la scène provoquant une erreur numérique perceptible humainement.

1.2 Contexte de la thèse

Le sujet de thèse présenté ici s'intitule « Détection de bruit dans les images stéréoscopiques par méthodes d'apprentissage automatique ». La thèse s'inscrit dans un projet ANR donc l'objectif final serait de pouvoir concevoir un moteur rendu mixte mêlant rendu temps réel (une image calculée en moins de 30 ms) et photo-réalisme. C'est à dire cibler la partie à améliorer par rapport au regard de l'utilisateur dans un support de rendu stéréoscopique. L'objectif de la thèse est double, à la fois détecter le bruit, le quantifier à partir de l'image uniquement et non les informations de la scène (si possible) puis proposer une méthode de filtrage/débruitage (post-traitement) permettant d'améliorer son aperçu. Tout en gardant pour objectif que cela puisse être réalisé en temps réel.

1.3 Problématique abordée

Comme évoqué précédemment, l'objectif de la thèse est double pouvoir à la fois détecter le bruit dans une image de synthèse issue d'un moteur de rendu et

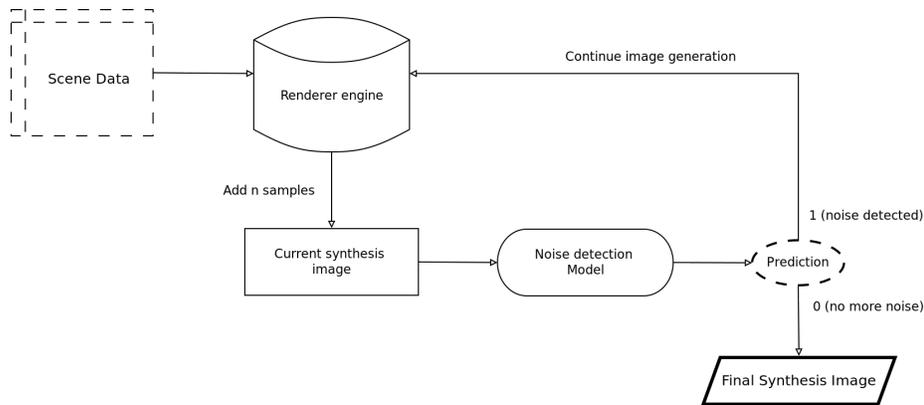


FIGURE 1.1 – Représentation de l’interaction du modèle de détection souhaité de bruit lors du rendu d’une image de synthèse à partir des données de la scène 3D

mais également proposer une méthode de réduction de ce bruit perçu (denoising method).

La Fig. 1.1 détaille davantage la manière dont le problème est abordé et surtout elle permet de cibler la partie sur laquelle la problématique de la thèse s’incruste. Nous avons donc le modèle de détection de bruit (modèle de classification binaire) qui permettrait de prédire si le moteur de rendu a encore la nécessité d’améliorer l’image ou non. On pourrait bien entendu ici, mettre en place une étape de post-traitement pour amélioration de l’image avant évaluation du bruit contenu dans celle-ci.

Chapitre 2

Bibliography

2.1 Bibliographie

Découper ici la bibliographie en sous sections :

- Algorithmes de rendus
- Image quality metrics
- Approches Deep learning pures
- Application détection de bruits images de synthèse
- SVD

Chapitre 3

Travaux réalisés

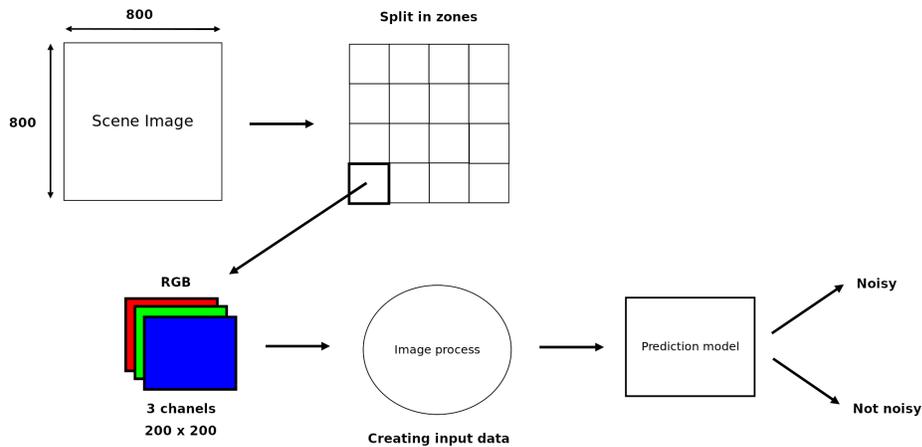


FIGURE 3.1 – Découpage de l’image d’une scène pour extraction des caractéristiques des zones de celle-ci

3.1 La base de données d’images

Cette partie sera consacrée à la présentation des différents travaux réalisés dans le cadre de la première année de thèse. La majorité des travaux ne donnant pas de résultats unanime, chaque approche sera donc détaillée pour mettre en avant ses avantages, ses inconvénients, les raisons pour lesquelles l’approche n’est pas forcément concluante pour enfin s’ouvrir à de nouveaux axes et méthodes d’applications.

Pour l’ensemble des travaux réalisés durant cette première année, l’image de taille 800×800 a été découpée en 16 zones de 200×200 (voir Fig. 3.1). C’est ensuite une zone qui est traitée pour extraction des informations caractérisant le bruit présent dans le celui-ci. La première partie de la thèse porte donc sur cette étape du processus de classification.

Une base de données d’images a été développée, elle contient 9 scènes ainsi que les seuils perceptifs prelevés lors d’expérience pour chaque zone d’une scène. Chaque sujet de l’expérience devait partir de l’image la plus bruitée augmenter la qualité (nombre d’échantillons) de chaque zone pour se rapprocher au maximum de l’image de référence (image obtenue avec le plus grand nombre d’échantillons). La Fig. 3.2 présente la manière dont chaque zone des différentes images (nombre d’échantillons) sont labélisés.

Dans le cadre de ses premières expériences, uniquement les scènes issues du moteur de rendu Maxwell ont été sélectionnées :

- Apart1opt02
- Bureau1
- SdbCentre
- SdbDroite

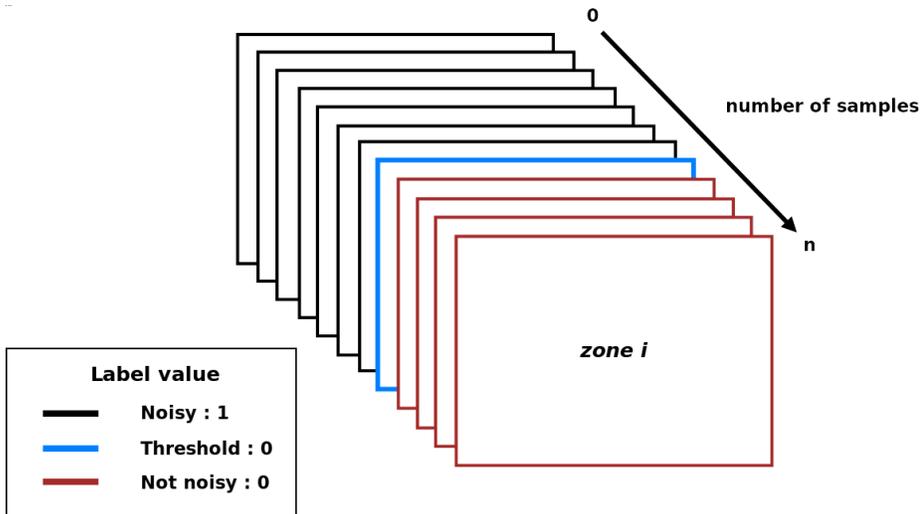


FIGURE 3.2 – Le seuil perceptif est identifié par la moyenne des scores obtenus sur les sujets de l’expérience. Les images situées avant ce seuil perceptif sont considérées comme bruitées, celles situées après, comme non bruitées.

Le processus de génération des données utilisées pour création des bases d’apprentissage, de validation et de test pour le modèle mathématique choisi est détaillé dans la Fig. 3.3. Ce processus vient de l’hypothèse que chaque zone est traitée indépendamment de la scène. Plutôt que d’utiliser 3 scènes en apprentissage et une en test, la sélection de z zones pour former la base d’apprentissage sont réalisées aléatoirement avec $z \in [4, 6, 8, 10, 12]$. Les zones restantes seront alors utilisées pour les bases de validation et de test du modèle.

Afin de mesurer l’importance de la séparabilité des données à partir du seuil, 3 manières de construire les données finales ont été abordées :

- **all** : consistant à prendre l’ensemble des données.
- **center** : consistant à ne prendre que les images à échantillons proche de 150 du seuil estimé de la zone.
- **split** : les données non sélectionnées par la seconde (données fortement extérieur au seuil).

Enfin, la comparaison des performances de modèles se fera via le score AUC - ROC (Area Under The Curve Receiver Operating Characteristics) sur la base de test. Il permet de mesurer la performance d’un modèle pour un problème de classification, plus particulièrement à quel point il arrive à bien séparer les classes du problème.

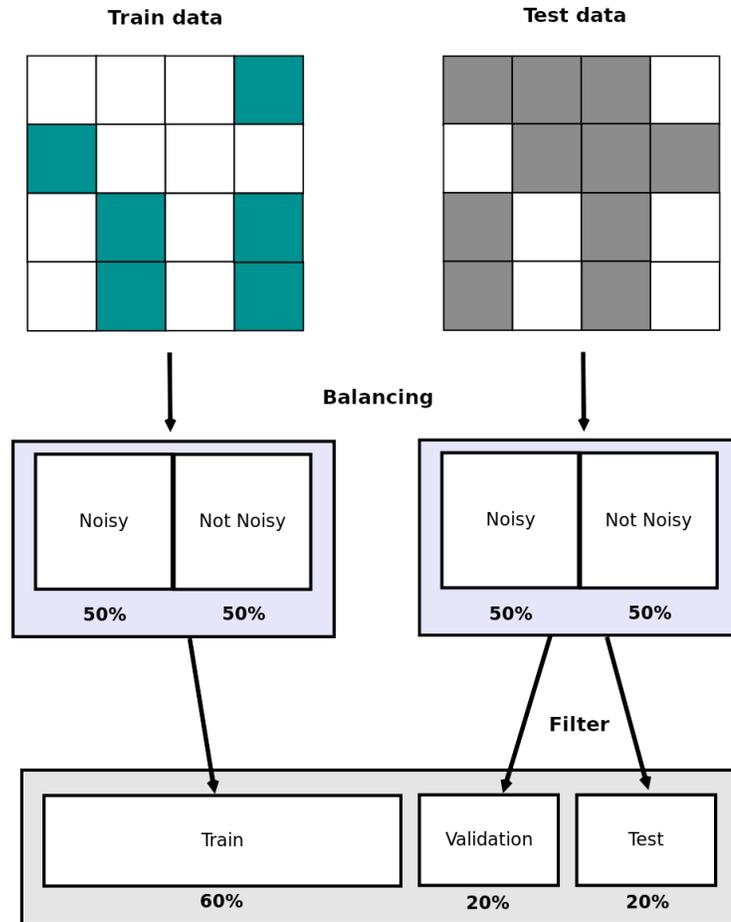


FIGURE 3.3 – Dans cet exemple, 6 zones ont été sélectionnées pour la base d'apprentissage. Un équilibrage des classes est réalisé pour ne pas biaiser le modèle lors de l'apprentissage puis la répartition des différentes bases est effectuée.

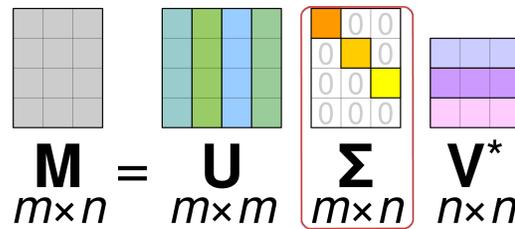


FIGURE 3.4 – Méthode de factorisation utilisée pour réduction de dimension de l'image : décomposition en valeurs singulières

3.2 L'approche décomposition SVD

3.2.1 Description de la méthode

La décomposition en valeurs singulières (SVD) est une méthode de factorisation de matrice. L'utiliser dans le cadre de la réduction d'informations pour le traitement de l'image pourrait être intéressant. La Fig 3.4 propose un aperçu de la factorisation proposée par la méthode.

La méthode propose donc une factorisation de la matrice de la manière suivante :

$$M = U \times \Sigma \times V^*$$

C'est le vecteur de valeurs singulières (Σ) auquel nous nous sommes principalement intéressé. En effet, la méthode de décomposition appliquée à une image réduite au canal de luminance L de la transformation L^*a*b (voir annexe A) permet d'obtenir le vecteur de valeurs singulières Σ .

Pour une image de 200×200 d'une zone nous obtenons donc un vecteur de valeurs singulières de taille 200. C'est ce vecteur que nous allons traiter par la suite.

Pourquoi s'être intéressé à la décomposition SVD ? Nous avons pu observer que suivant le nombre d'échantillons utilisés pour générer une image d'un scène, le vecteur de valeurs singulières semblaient significativement bien distinguer le niveau de bruit jusqu'à l'image de référence (voir Fig. 3.5).

3.2.2 Sélection des attributs

List des attributs essayés

Approche dites naïves de sélection d'attributs

Autres approches de sélection

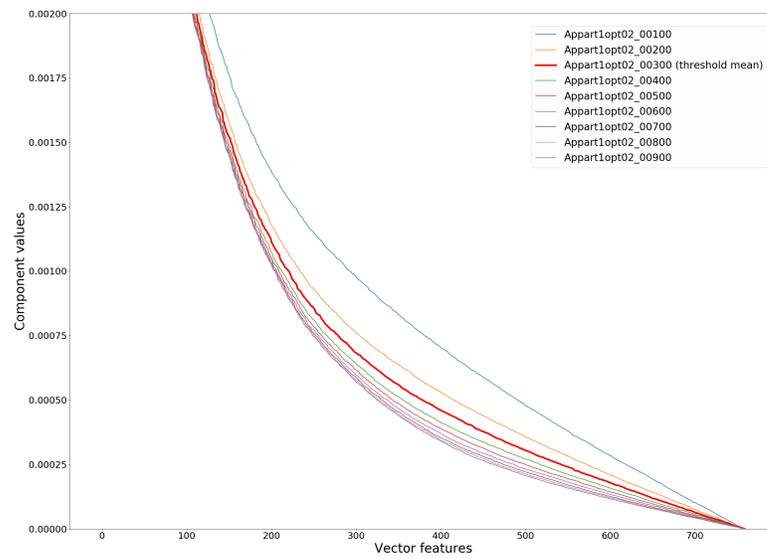


FIGURE 3.5 – Aperçu du vecteur de valeurs singulières des images de la scène Appart1opt02 à différents niveaux de bruits. Ici le vecteur possède 800 valeurs car la décomposition a été appliquée sur l'ensemble de l'image.

3.2.3 Étude des échantillons de rendu

3.2.4 Les approches futures

Chapitre 4

Enseignements

- **L1 Mathématiques** : Fondements de l'algorithmique (19.38 Eq TD)
- **L2 Informatique** : Algorithmique avancée en C++ (12 Eq TD)
- **DUT 1^{re} année Informatique** : Bases de la POO (28 Eq TD)

Chapitre 5

Formation

5.1 Conclusion

Appendices

Annexe A

Transformation L^*a^*b

L*a*b est obtenu en utilisant la transformation XYZ de la matrice d'image (espace CIZ XYZ). Les équations (A.1), (A.2), (A.3) détaillent la manière dont chaque canal est calculé.

$$L = 116f_y - 16 \quad (\text{A.1})$$

$$a = 500(f_x - f_y) \quad (\text{A.2})$$

$$b = 200(f_y - f_z) \quad (\text{A.3})$$

où

$$f_x = \begin{cases} \sqrt[3]{x_r} & \text{if } x_r > \epsilon \\ \frac{\kappa x_r + 16}{116} & \text{otherwise} \end{cases} \quad (\text{A.4})$$

$$f_y = \begin{cases} \sqrt[3]{y_r} & \text{if } y_r > \epsilon \\ \frac{\kappa y_r + 16}{116} & \text{otherwise} \end{cases} \quad (\text{A.5})$$

$$f_z = \begin{cases} \sqrt[3]{z_r} & \text{if } z_r > \epsilon \\ \frac{\kappa z_r + 16}{116} & \text{otherwise} \end{cases} \quad (\text{A.6})$$

$$x_r = \frac{X}{X_r}, \quad y_r = \frac{Y}{Y_r}, \quad z_r = \frac{Z}{Z_r}$$

$$\epsilon = \begin{cases} 0.008856 & \text{Actual CIE standard} \\ 216/24389 & \text{Intent of the CIE standard} \end{cases} \quad (\text{A.7})$$

$$\kappa = \begin{cases} 903.3 & \text{Actual CIE standard} \\ 24389/27 & \text{Intent of the CIE standard} \end{cases} \quad (\text{A.8})$$

Annexe B

Autre annexe