

CSC_4IM01_TP : Introduction au Traitement des Images

Restauration d'Images Sous-marines

**Fouad KHELIFI
Hortense BONNEAU**

Table des matières

| | |
|---|-----------|
| 1 État de l'Art du Traitement des Images Sous-Marines | 3 |
| 1.1 Introduction | 3 |
| 1.2 Méthodes de Restauration d'Image | 3 |
| 1.3 Méthodes d'Amélioration d'Image | 3 |
| 1.4 Méthodes par Apprentissage | 3 |
| 2 Correction des Couleurs à l'Aide de l'Espace RGB | 4 |
| 2.1 Hypothèse du Monde Gris | 4 |
| 2.2 Hypothèse des White Patches | 5 |
| 3 Correction des Couleurs à l'Aide de l'Espace $l\alpha\beta$ | 7 |
| 3.1 Conversion dans l'Espace $l\alpha\beta$ | 7 |
| 3.2 Correction et Amélioration dans l'Espace $l\alpha\beta$ | 8 |
| 3.2.1 Hypothèse du Monde Gris | 8 |
| 3.2.2 Hypothèse des White Patches | 9 |
| 3.2.3 Égalisation d'Histogramme sur le Canal l | 9 |
| 4 Méthode ACE : Égalisation Automatique des Couleurs | 10 |
| 4.1 Modèle et Formulation Mathématique | 11 |
| 4.2 Fonction Pente Polynomiale | 11 |
| 4.2.1 Gestion des Bords de l'Image | 11 |
| 4.2.2 Approximation Polynomiale | 12 |
| 4.2.3 Calcul avec les Convolutions | 12 |
| 4.3 Implémentation pour la Correction des Images Sous-marines | 12 |
| 4.3.1 Test de l'ACE sur des Images Synthétiques | 12 |
| 4.3.2 Correction des Images Sous-marines avec l'ACE | 13 |
| 4.3.3 Application de l'ACE sur le Canal l | 13 |
| 5 Conclusion | 14 |
| Table des figures | 15 |
| Références | 16 |

1 État de l'Art du Traitement des Images Sous-Marines

1.1 Introduction

Le traitement des images sous-marines est crucial pour surmonter les limitations inhérentes à l'environnement aquatique, comme la faible visibilité, le flou, et la dominante bleue causée par l'absorption et la diffusion de la lumière. Ces défis limitent la portée visuelle et la qualité des images, en particulier à de grandes profondeurs ou dans des eaux troubles. Les méthodes proposées pour résoudre ces problèmes se répartissent en deux catégories principales : **la restauration d'image** et **l'amélioration d'image**, chacune ayant des applications et des contraintes spécifiques.

1.2 Méthodes de Restauration d'Image

Les images sous-marines subissent une dégradation importante causée par l'absorption et la diffusion de la lumière. L'absorption réduit progressivement l'énergie lumineuse, supprimant les couleurs selon leur longueur d'onde (par exemple, le rouge disparaît après environ 3 mètres). La diffusion, quant à elle, provoque un flou et diminue le contraste, amplifiant le *voile* visuel. Les techniques de restauration compensent ces effets par :

- **Modélisation physique** : Exploite les propriétés optiques de l'eau (absorption, diffusion) et des fonctions de transfert pour corriger flous et pertes de contraste.
- **Déconvolution** : Applique des filtres, comme celui de Wiener, basés sur des fonctions mesurant la propagation lumineuse pour restaurer les détails.
- **Analyse de polarisation** : Isolant la lumière diffusée grâce à des filtres polarisants, elle réduit les voiles optiques et améliore le contraste.

Ces méthodes, bien que précises, dépendent fortement des caractéristiques de l'environnement aquatique.

1.3 Méthodes d'Amélioration d'Image

Les techniques d'amélioration visent à optimiser la qualité visuelle des images sans nécessiter de modèles physiques complexes. Elles se concentrent sur la correction des couleurs, l'ajustement des contrastes et la suppression des dominantes chromatiques.

- **Espace $la\beta$** : En convertissant les images dans un espace de couleur perceptuel, cette approche ajuste indépendamment la luminosité, la teinte et la saturation pour atténuer la dominante bleue et rehausser les détails visuels.
- **Automatic Color Equalization (ACE)** : Inspirée de la perception humaine, cette méthode ajuste automatiquement les couleurs et le contraste pour compenser les dominantes dues à l'absorption sous-marine, offrant des images plus équilibrées et naturelles.

Le document met en avant et implémente spécifiquement ces deux méthodes, en raison de leur efficacité et de leur capacité à traiter rapidement des images sous-marines tout en améliorant leur rendu visuel. L'intégralité du code source ainsi que les images utilisées dans ce projet sont disponibles sur le dépôt GitHub accessible via [ce lien](#).

1.4 Méthodes par Apprentissage

Il existe plusieurs méthodes par apprentissage qui se proposent de restaurer des images sous-marines, on peut citer par exemple :

- **Underwater Image Restoration via Contrastive Learning and a Real-World Dataset** : la restauration d'image sous-marines par apprentissage contrastif tout en construisant une base de données d'images sur lesquelles s'appuyer pour entraîner l'algorithme. La méthode de transfert d'image à image non-supervisée repose sur de l'apprentissage contrastif en le combinant avec des réseaux antagonistes génératifs (GAN). Le but étant de maximiser l'information mutuelle entre patchs correspondants à l'image brute, et à l'image

restaurée afin de pouvoir saisir les correspondances de contenu et les similarités de couleur entre les deux domaines d'image.

Le concept de GAN est constitué de deux éléments : le générateur G et le discriminateur D . Tandis que le générateur s'en tient à générer des images les plus réalistes possibles, le discriminateur lui, est chargé d'évaluer si les données générées sont réelles ou fausses. Comme les deux réseaux sont en compétition, cela permet d'améliorer progressivement la qualité des données générées.

La base de données d'image est divisée en trois parties : des images de basse qualité, des images de haute qualité et des images restaurées. Les images de bonne qualité sont associées à leur version restaurée et constituent l'entrée et la référence d'un premier modèle supervisé, aussi appelé *apprentissage par paires*. Les images de plus basse qualité constituent l'entrée d'un second modèle non-supervisé, cette fois avec comme référence d'images réelles, le set d'images restaurées. Ce type d'apprentissage est dit *sans paires*.

Afin d'effectuer le mappage, c'est-à-dire le processus de reconnaissance des correspondances d'un ensemble de données à un autre, on extrait les caractéristiques des images identifiées depuis différentes couches de l'encodeur du générateur que l'on redirige dans un Perceptron à deux couches afin de pouvoir projeter ces caractéristiques dans un nouvel espace.

La méthode se propose de minimiser une fonction de perte générale, résultat de l'addition de trois fonctions de pertes, pondérées par des paramètres.

- La première fonction de perte est celle de la perte antagoniste Adversarial Loss : elle permet de produire des images plus réalistiques.
- La seconde fonction est celle de PatchNCE Loss : cette dernière vise à maximiser l'information mutuelle entre les patchs correspondants entre entrée et sortie du générateur.
- Enfin la dernière est la fonction identité de perte, Identity Loss, qui encourage le générateur à éviter tout changement de structure ou de couleur non-nécessaire entre l'entrée et la sortie.

Ensuite les deux modèles entraînés sont utilisés pour l'évaluation de l'algorithme.

2 Correction des Couleurs à l'Aide de l'Espace RGB

2.1 Hypothèse du Monde Gris

C'est l'hypothèse selon laquelle l'image devrait avoir, en moyenne, une couleur grise. C'est-à-dire que les moyennes des trois canaux de couleur (R, G, B) doivent être égales pour une scène correctement éclairée. Si ce n'est pas le cas, la correction de couleur va réajuster les valeurs de chaque canal pour compenser les biais de couleur.

L'idée de base de la correction est que, sous un éclairage uniforme, la somme des intensités des trois canaux (R, G, B) dans l'image devrait être équivalente, car la moyenne des couleurs d'une scène "idéale" sous lumière neutre devrait tendre vers le gris (même intensité dans les trois canaux).

Cette méthode est implémentée via la fonction `process_underwater_image` qui est présente dans le fichier `Undermarine_lib.py`, de la manière suivante :

1. Calculer la moyenne de chaque canal (R, G, B) de l'image :

$$\bar{R} = \frac{1}{N} \sum_{x=1}^N R(x) \quad \bar{G} = \frac{1}{N} \sum_{x=1}^N G(x) \quad \bar{B} = \frac{1}{N} \sum_{x=1}^N B(x)$$

où N est le nombre de pixels dans l'image, et $R(x), G(x), B(x)$ sont les valeurs RGB du pixel x .

2. Calculer la moyenne de l'illuminant (moyenne des trois moyennes des canaux) :

$$\bar{M} = \frac{\bar{R} + \bar{G} + \bar{B}}{3}$$

3. Ajuster les canaux pour égaliser les moyennes : La correction consiste à ajuster les valeurs des canaux pour qu'ils aient tous la même moyenne. Pour chaque pixel, chaque composante de couleur (R, G, B) est divisée par sa moyenne respective et multipliée par la moyenne de l'illuminant :

$$R_{\text{corr}}(x) = R(x) \times \frac{\bar{M}}{R} \quad G_{\text{corr}}(x) = G(x) \times \frac{\bar{M}}{G} \quad B_{\text{corr}}(x) = B(x) \times \frac{\bar{M}}{B}$$



FIGURE 2.1 – Résultats de la correction RGB (avec l'hypothèse du monde gris) sur des images sous-marines

Comme illustré dans la FIGURE 2.1 (le reste des images traitées sont dans le fichier `Images_RGB_GW`), pour les images ayant une composition variée et équilibrée, l'hypothèse du monde gris fonctionne correctement. Cependant, pour les images dominées par une couleur unique, comme celles contenant une grande étendue d'eau, la correction devient incorrecte. Sous l'eau, la lumière est principalement bleue, et l'algorithme, en cherchant à équilibrer les canaux, exagère la composante rouge. Cela résulte en une image où les objets hors de l'eau, comme les rochers ou les poissons, apparaissent avec une dominante rouge excessive, ce qui montre les limites de l'hypothèse du monde gris.

2.2 Hypothèse des White Patches

L'hypothèse des White Patches repose sur l'idée que certaines zones de l'image devraient être proches du blanc, même sous un éclairage non uniforme. En ajustant ces zones lumineuses (les "patchs blancs") pour qu'elles correspondent à un blanc idéal, on peut corriger les dominantes de couleur. Sur le principe on identifie les zones les plus lumineuses de l'image et on les ajuste pour qu'elles correspondent à un point blanc idéal [255, 255, 255]. Cela permet de corriger les dominantes de couleur sans affecter les autres zones de l'image.

Cette méthode est implémentée via la fonction `process_underwater_image` qui est présente dans le fichier `Undermarine_lib.py`, de la manière suivante :

- Conversion en niveaux de gris :** L'image est d'abord convertie en niveaux de gris pour identifier les zones les plus lumineuses noté $\text{Gray}(x)$
- Identification des patchs lumineux :** On sélectionne les $p\%$ des pixels les plus lumineux (par exemple, 2% de l'image) comme étant les "patchs blancs" :

$$P_{\text{bright}} = \{x \in \{1, \dots, N\} \mid \text{Gray}(x) \geq \theta\}$$

Où :

- P_{bright} représente l'ensemble des indices des pixels lumineux.
- θ est un seuil qui détermine la luminosité maximale pour avoir $\#P_{\text{bright}} = \frac{p}{100} \times \#\text{Gray}$

3. Calcul de la moyenne des patchs lumineux : La moyenne des couleurs des patchs lumineux est calculée :

$$\bar{R}_{\text{bright}} = \frac{1}{\#P_{\text{bright}}} \sum_{x \in P_{\text{bright}}} R(x) \quad \bar{G}_{\text{bright}} = \frac{1}{\#P_{\text{bright}}} \sum_{x \in P_{\text{bright}}} G(x) \quad \bar{B}_{\text{bright}} = \frac{1}{\#P_{\text{bright}}} \sum_{x \in P_{\text{bright}}} B(x)$$

4. Application de la correction : Un facteur de correction est appliqué à l'image pour ajuster les canaux R , G , et B qui est un réajustement du point blanc idéal.

$$R_{\text{corr}}(x) = R(x) \times \frac{255}{\bar{R}_{\text{bright}}} \quad G_{\text{corr}}(x) = G(x) \times \frac{255}{\bar{G}_{\text{bright}}} \quad B_{\text{corr}}(x) = B(x) \times \frac{255}{\bar{B}_{\text{bright}}}$$

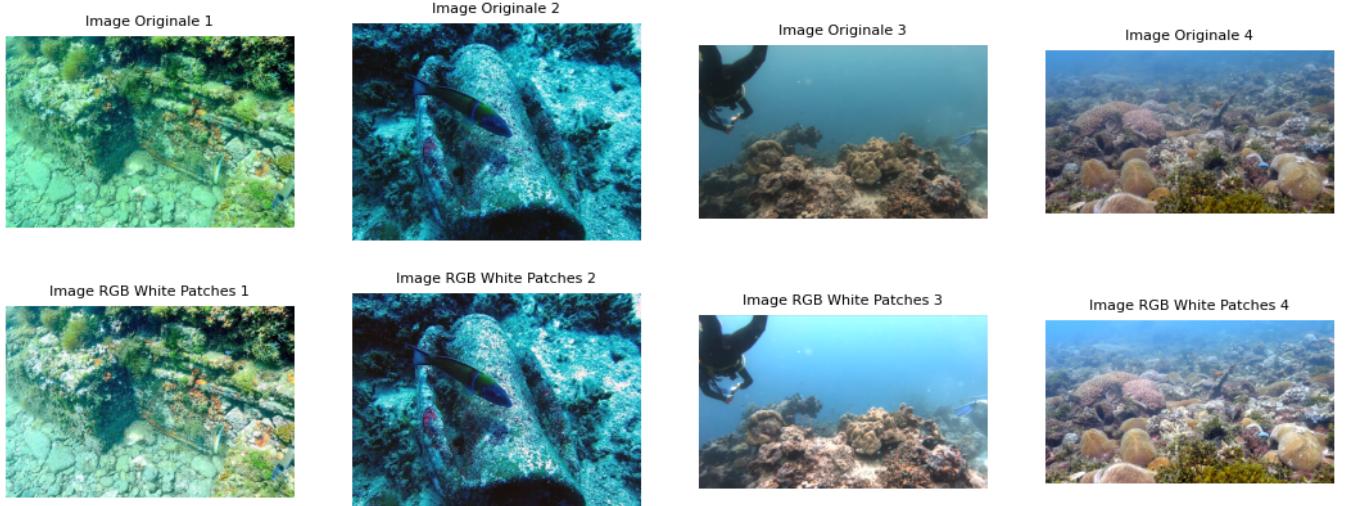


FIGURE 2.2 – Résultats de la correction RGB (avec l'hypothèse des white patches avec $p = 2\%$)

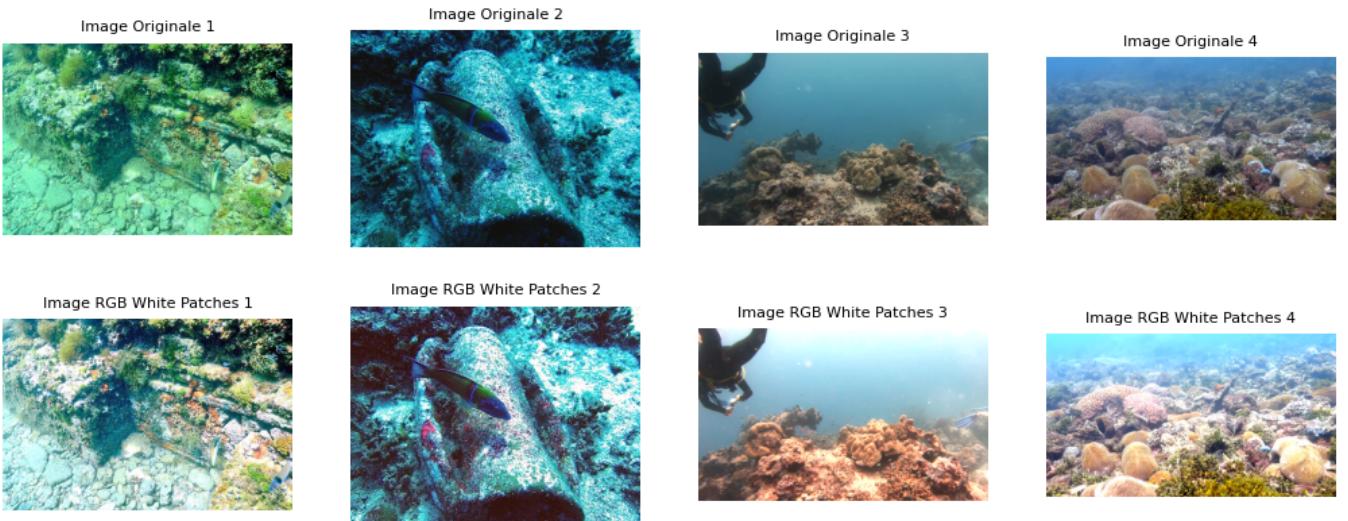


FIGURE 2.3 – Résultats de la correction RGB (avec l'hypothèse des white patches avec $p = 20\%$)

Dans la FIGURE 2.2 (le reste des images traitées sont dans le fichier `Images_RGB_WP_2p`), les deux premières images n'ont pas été corrigées, car l'hypothèse des patchs blancs n'est pas pertinente ici : bien qu'il y ait un voile bleu, plusieurs zones blanches sont présentes, ce qui fausse l'application de cette méthode. Cependant, comme illustré dans la FIGURE 2.3 (le reste des images traitées sont dans le fichier `Images_RGB_WP_20p`), en augmentant la valeur de p , l'hypothèse des white patches devient progressivement valide même pour ces deux premières

images, car un plus grand nombre de pixels lumineux est pris en compte, ce qui permet de capturer davantage de zones lumineuses pertinentes. Cette augmentation améliore la correction des dominantes colorées et permet d'obtenir des résultats proches de ceux de l'hypothèse du monde gris.

Cependant, augmenter excessivement la valeur de p peut engendrer des effets indésirables. En incluant trop de pixels dans la sélection des zones lumineuses, le facteur de correction devient biaisé par des zones qui ne devraient pas contribuer au calcul du point blanc. Cela peut entraîner une trop grande augmentation de la luminosité globale de l'image, rendant certaines zones surexposées et moins naturelles. Il est donc essentiel de choisir une valeur optimale de p qui équilibre la correction des couleurs et la préservation de la dynamique lumineuse de l'image.

3 Correction des Couleurs à l'Aide de l'Espace $l\alpha\beta$

Cette méthode repose sur une transformation dans l'espace colorimétrique perceptuel $l\alpha\beta$, qui un modèle de couleurs standardisé par la Commission Internationale de l'Éclairage (CIE) en 1976. Contrairement aux espaces colorimétriques comme RGB, qui sont basés sur des caractéristiques physiques de la lumière ou des pigments, l'espace $l\alpha\beta$ cherche à décrire les couleurs de manière plus uniforme et perceptuellement linéaire, ce qui signifie que les différences entre les couleurs dans cet espace correspondent mieux à ce que l'œil humain perçoit.

3.1 Conversion dans l'Espace $l\alpha\beta$

Les étapes de conversion décrites ci-dessous ont été implémentées dans la fonction `process_underwater_image` du programme `Undermarine_lib.py`. On considère le vecteur $F(x) = [R(x), G(x), B(x)]^T$, défini au pixel x , où $R(x)$, $G(x)$, et $B(x)$ sont les valeurs des trois canaux RGB normalisées dans l'intervalle $[0, 1]$.

- Correction gamma :** Les valeurs RGB brutes issues du capteur ne représentent pas directement les intensités lumineuses physiques, car elles sont affectées par une courbe gamma non linéaire appliquée pour optimiser le rendu visuel. Alors une correction gamma inverse au gamma appliqué lors de la capture ou du rendu de l'image est appliquée :

$$F'(x) = F(x)^{\frac{1}{\gamma}}$$

- Conversion RGB → XYZ :** Les valeurs RGB corrigées sont transformées dans l'espace XYZ, une représentation standard des couleurs perceptibles par l'œil humain :

$$X(x) = T_{xyz} \cdot F'(x)$$

où T_{xyz} est une matrice de conversion prédéfinie et standardisée par la CIE.

- Conversion XYZ → LMS :** L'espace LMS représente les réponses des cônes de l'œil humain (Long, Medium, Short). La transformation est donnée par :

$$L(x) = T_{lms} \cdot X(x)$$

avec T_{lms} la matrice de conversion associée.

- Transformation logarithmique :** Les valeurs LMS sont transformées en appliquant une fonction logarithme :

$$L_{\log,j}(x) = \log(L(x))$$

- Décorrélation via PCA pour obtenir $l\alpha\beta$:** Une analyse en composantes principales (PCA) est appliquée pour séparer la luminance et les composantes chromatiques. Cette transformation est donnée par :

$$L_{l\alpha\beta}(x) = T_{pca} \cdot L_{\log}(x),$$

où T_{pca} est une matrice de décorrélation.

Ainsi, nous avons obtenu les coordonnées du vecteur $L_{l\alpha\beta}(x)$ de l'image dans l'espace $l\alpha\beta$. Les trois axes principaux résultants sont orthogonaux, et il est constaté que le premier axe l représente un canal achromatique, tandis que les deux autres canaux (α et β) sont des canaux chromatiques jaune-bleu et rouge-vert, respectivement :

- $l \propto (r + g + b)$,
- $\alpha \propto (r + g - b)$,
- $\beta \propto (r - g)$.

Ainsi, dans cet espace, les composantes de couleur sont découplées de la luminance.

3.2 Correction et Amélioration dans l'Espace $l\alpha\beta$

Dans l'espace $l\alpha\beta$, la correction des couleurs se fait en ajustant les canaux α et β , qui représentent respectivement les variations chromatiques jaune-bleu et rouge-vert. Deux méthodes de correction sont appliquées : l'hypothèse du **monde gris** et l'hypothèse des **white patches**.

3.2.1 Hypothèse du Monde Gris

L'idée est que la moyenne des valeurs de α et β dans une scène équilibrée doit être nulle, permettant ainsi de supprimer les dominantes de couleur. La correction se fait en soustrayant la moyenne des valeurs des canaux α et β :

$$\alpha^*(x) = \alpha(x) - \bar{\alpha} \quad \beta^*(x) = \beta(x) - \bar{\beta}$$

où $\bar{\alpha}$ et $\bar{\beta}$ sont les moyennes des canaux α et β , respectivement, cette méthode est implémentée via la fonction `gray_world_lab` qui est présente dans le fichier `Undermarine_lib.py`.

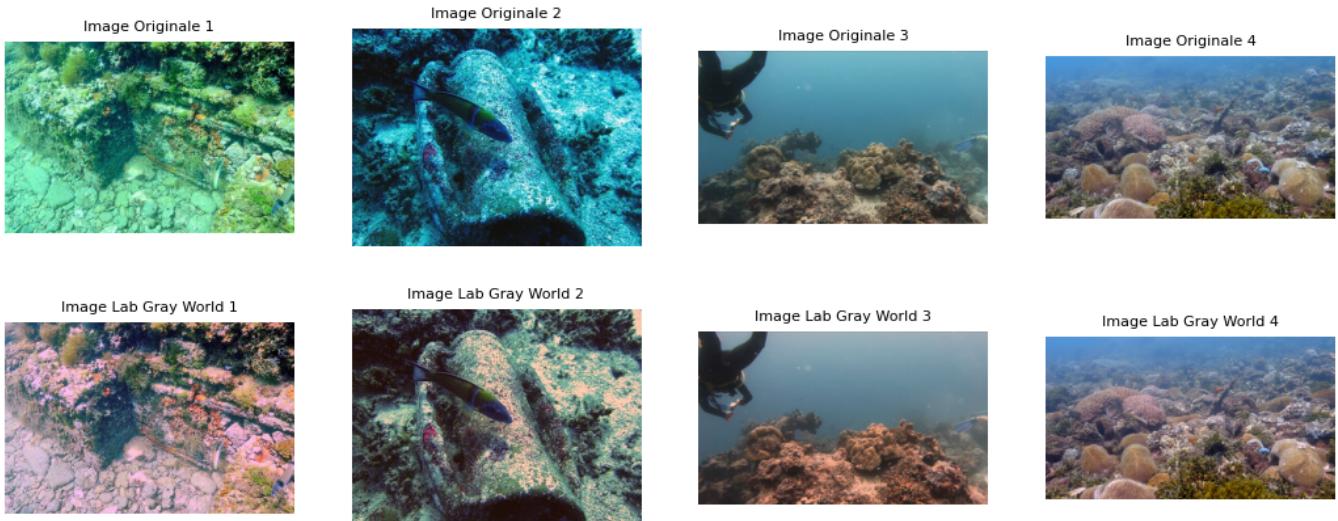


FIGURE 3.1 – Résultats de la correction $l\alpha\beta$ (avec l'hypothèse du monde gris) sur des images sous-marines

Comme on peut l'observer sur la FIGURE 3.1 (le reste des images traitées sont dans le fichier `Images_Lab_GW`), la correction des couleurs dans l'espace $l\alpha\beta$ est nettement meilleure que dans l'espace RGB. Cela s'explique par le fait que dans $l\alpha\beta$, la luminance (l) est séparée des composantes chromatiques (α et β). Cette séparation permet d'effectuer des ajustements spécifiques sur les canaux sans altérer la luminosité globale de l'image.

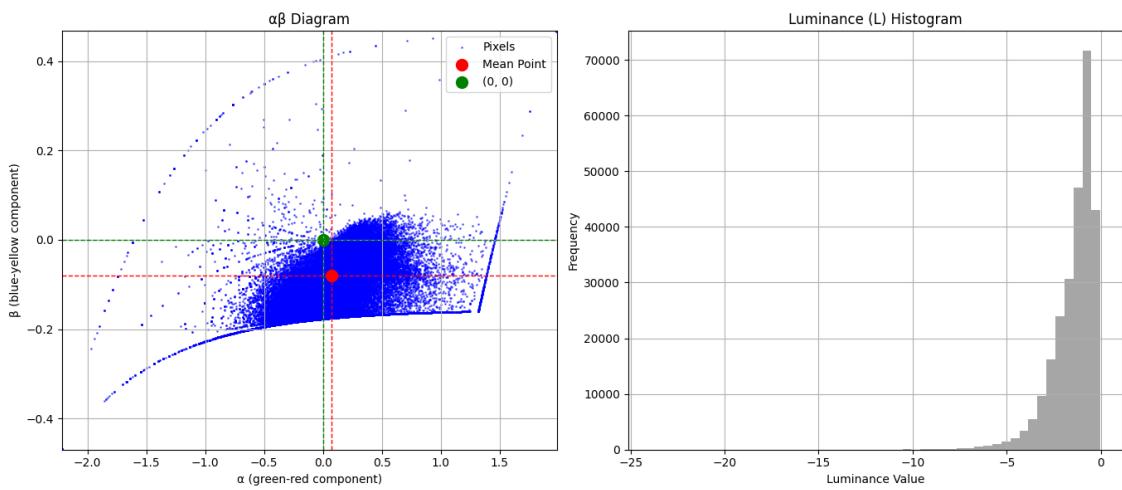


FIGURE 3.2 – Diagramme $\alpha\beta$ et Histogramme de la Luminance avant la correction

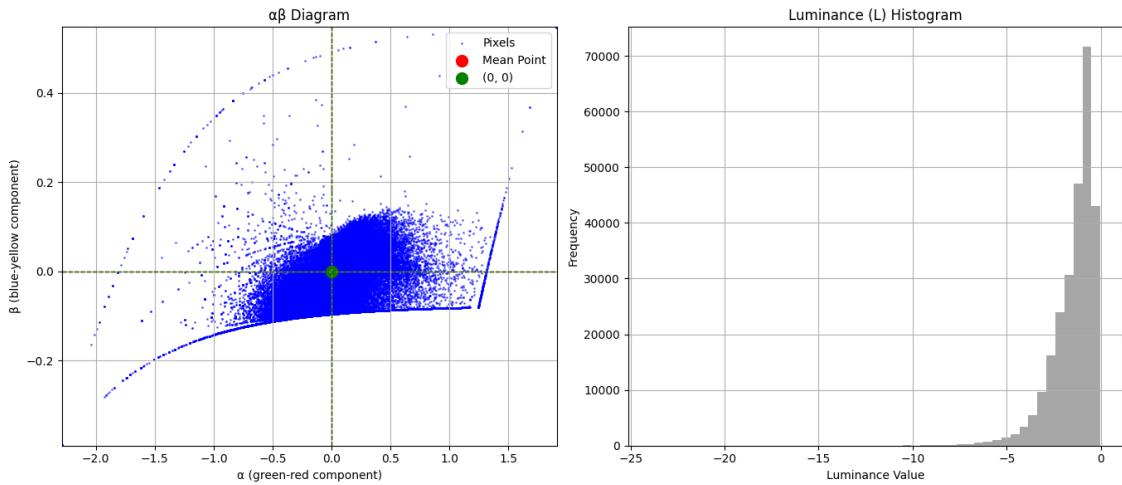


FIGURE 3.3 – Diagramme $\alpha\beta$ et Histogramme de la Luminance après la correction

3.2.2 Hypothèse des White Patches

Cette méthode ajuste les canaux α et β en fonction des pixels les plus lumineux (les "patchs blancs"). On identifie les $p\%$ des pixels les plus lumineux, puis on calcule la moyenne des canaux α et β pour ces pixels lumineux :

$$\alpha^*(x) = \alpha(x) - \bar{\alpha}_{\text{bright}} \quad \beta^*(x) = \beta(x) - \bar{\beta}_{\text{bright}}$$

où $\bar{\alpha}_{\text{bright}}$ et $\bar{\beta}_{\text{bright}}$ sont les moyennes des canaux α et β pour les pixels lumineux, identifiés en fonction de leur luminance.

La même observation s'applique qu'à la section 2.2 : si l'image d'origine contient des pixels naturellement blancs, elle ne satisfait pas l'hypothèse des patchs blancs, rendant la correction inefficace. En revanche, lorsque l'hypothèse est vérifiée, les résultats obtenus sont comparables à ceux de l'hypothèse du monde gris (les images traitées sont dans les fichiers [Images_Lab_WP_2p](#) et [Images_Lab_WP_20p](#)).

3.2.3 Égalisation d'Histogramme sur le Canal l

Pour améliorer le contraste des images corrigées, une égalisation d'histogramme a été appliquée au canal l des images traitées selon l'hypothèse du monde gris. Cette opération vise à redistribuer les valeurs de luminance de manière à mieux occuper la plage dynamique disponible.

Soit $l : \Omega \rightarrow [0, 1]$, où Ω est le domaine de l'image et les intensités du canal l sont normalisées dans l'intervalle $[0, 1]$. L'égalisation d'histogramme est effectuée via le calcul de l'histogramme cumulé F , défini comme :

$$F(t) = \frac{1}{N} \#\{x \in \Omega : I(x) \leq t\}, \quad t \in [0, 1],$$

où N est le nombre total de pixels dans l'image. L'image égalisée est obtenue en appliquant F sur $I(x)$, ce qui donne :

$$I'(x) = F(I(x)).$$

Lorsque F est inversible, l'image résultante $I'(x)$ possède des intensités uniformément réparties, car :

$$P(F(I) \leq t) = F(F^{-1}(t)) = t.$$

En pratique, F n'est pas strictement inversible en raison de la quantification des intensités, mais l'image corrigée présente un histogramme approximativement uniforme. Cette méthode a permis d'améliorer la qualité visuelle des images corrigées comme le montre la FIGURE 3.4 (le reste des images traitées sont dans le fichier `Images_Lab_GW+_Eqalization`), mais cela fait ressortir plus les composantes rouges des images.

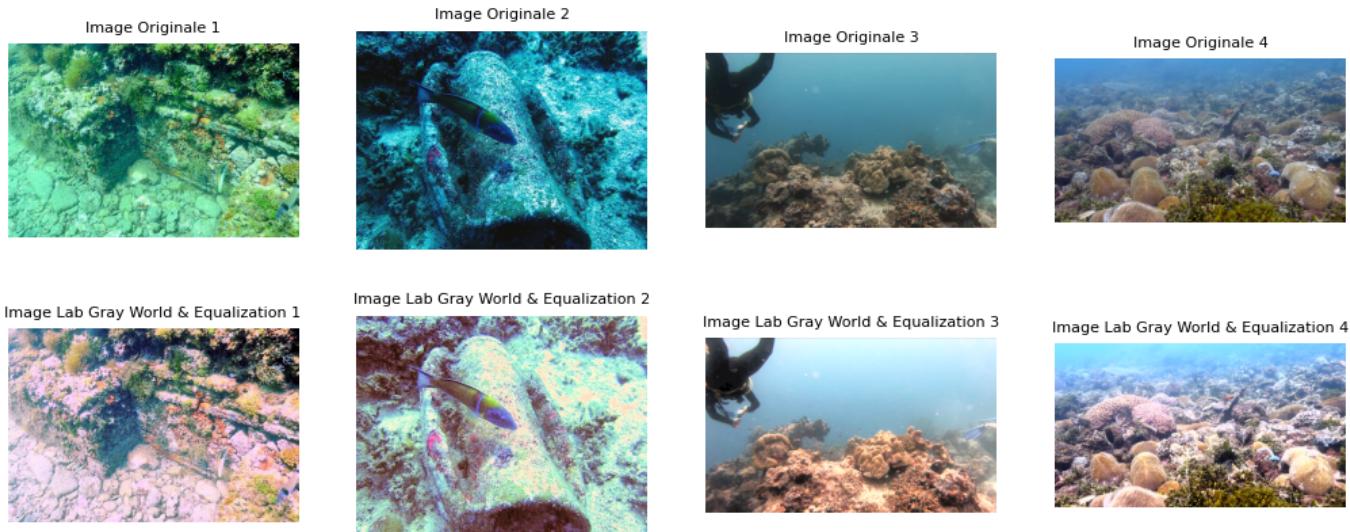


FIGURE 3.4 – Résultats de l'égalisation d'histogramme sur le canal l des images de la FIGURE 3.1

4 Méthode ACE : Égalisation Automatique des Couleurs

Une autre technique qui pourrait nous être utile pour la correction des images sous-marines est l'Automatic Color Equalization (ACE), une méthode conçue pour améliorer à la fois les couleurs globales et locales dans les images. Inspirée par les processus visuels humains, ACE combine deux fonctions essentielles : **la constance des couleurs**, qui assure une perception cohérente des teintes malgré les variations d'éclairage, et **l'égalisation de la luminosité**, qui ajuste dynamiquement les contrastes de l'image.

ACE fonctionne en calculant une nouvelle valeur pour chaque pixel en tenant compte des contributions des autres pixels, pondérées par leur distance spatiale et leur différence chromatique. En s'appuyant sur des approches traditionnelles de correction globale telles que **Gray World** et **White Patch**, ACE élimine efficacement les dominantes de couleur tout en maintenant un contraste équilibré, tant sur le plan global que local.

4.1 Modèle et Formulation Mathématique

L'algorithme ACE repose sur une formulation mathématique qui calcule une nouvelle valeur pour chaque pixel d'une image en tenant compte de la distribution spatiale et chromatique de tous les autres pixels. Pour chaque pixel x , la valeur corrigée $R(x)$ est donnée par :

$$R(x) = \sum_{y \in \Omega \setminus x} \frac{s_\alpha(I(x) - I(y))}{d(x-y)}, \quad x \in \Omega \quad (4.1)$$

où :

- $I(x)$ représente l'intensité du pixel x , avec $I : \Omega \longleftrightarrow [0, 1]$.
- $s_\alpha : [-1, 1] \rightarrow \mathbb{R}$ est une fonction de pente (slope function) qui amplifie ou comprime les différences locales selon le paramètre $\alpha \geq 1$.

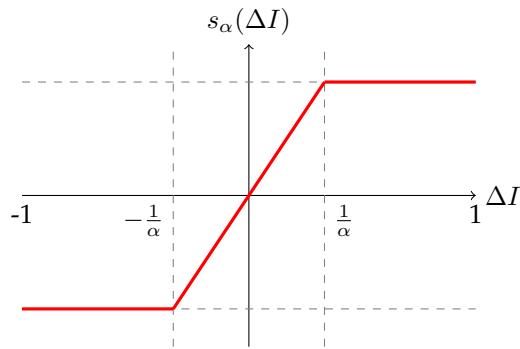


FIGURE 4.1 – Fonction pente s_α

- $d(x, y)$ est une fonction de distance spatiale, typiquement proportionnelle à la distance euclidienne entre x et y .

La pondération spatiale permet de moduler l'influence des pixels voisins en fonction de leur distance et de leur intensité relative. Cette combinaison garantit une correction équilibrée entre les ajustements globaux et locaux.

Après avoir calculé $R(x)$, les valeurs sont étirées pour occuper la plage dynamique complète de l'image, généralement entre 0 et 1, avec :

$$L(x) = \frac{R(x) - \min R}{\max R - \min R} \quad (4.2)$$

L'algorithme ACE, dans sa forme brute, présente une complexité élevée de $O(N^4)$, ce qui le rend impraticable pour des images de grande taille. Même pour des images relativement petites, le temps de calcul devient trop long lorsque cette méthode est utilisée en force brute. Il est donc nécessaire de trouver une autre façon d'implémenter cet algorithme afin de le rendre plus efficace.

4.2 Fonction Pente Polynomiale

Pour résoudre ce problème, la fonction de pente s_α est approximée par un polynôme, ce qui simplifie les calculs tout en conservant une précision suffisante. Cette approximation réduit la complexité de l'algorithme à $O(N^2 \log N)$ en utilisant la transformée discrète du cosinus (DCT) pour les convolutions.

4.2.1 Gestion des Bords de l'Image

Lors des convolutions sur une image, la gestion des bords est un défi. L'extension symétrique à demi-échantillon consiste à répliquer les pixels de bord de manière symétrique en miroir à l'extérieur de l'image pour éviter les artefacts causés par une simple périodisation ou un padding.

Donc si l'image est de taille $N \times M$ le nouveau domaine \mathbb{T}^2 qui fait référence à l'ensemble des positions de pixels dans l'image utilisées pour calculer $R(x)$, est un tore $(2N \times 2M)$ -périodique. L'équation (4.1) devient :

$$R(x) = \sum_{y \in \mathbb{T}^2} \omega(x - y) s_\alpha(I(x) - I(y)), \quad x \in \mathbb{T}^2 \quad (4.3)$$

où $\omega : \mathbb{T}^2 \rightarrow \mathbb{R}_+$ est une pondération définie par :

$$\omega(x - y) = \begin{cases} 0 & \text{si } x = y \\ \frac{1}{d(x - y)} & \text{si } x \neq y \end{cases}$$

4.2.2 Approximation Polynomiale

La fonction de pente s_α est approximée par un polynôme impair, ce qui simplifie les calculs tout en maintenant une précision suffisante. Cette approximation minimise l'erreur maximale sur l'intervalle $[-1, 1]$ en optimisant les coefficients c_m du polynôme.

$$\min_c \max_{t \in [-1, 1]} \left| s_\alpha(t) - \sum_{m=1}^D c_m t^m \right|$$

Les coefficients optimaux du polynôme pour différentes valeurs de α peuvent être obtenus via des méthodes d'optimisation telles que l'algorithme de Remez.

4.2.3 Calcul avec les Convolutions

Cette approximation polynomiale permet de décomposer $R(x)$ en une somme de convolutions :

$$R(x) = \sum_{n=0}^D a_n(I(x)) (\omega * I^n)(x) \quad (4.4)$$

où :

- $\omega * I^n$ est une convolution cyclique sur \mathbb{T}^2 .
- $a_n(t)$ un polynôme défini par :

$$\sum_{m=-n}^D c_m \binom{m}{n} (-1)^{m-n+1} t^{m-n}$$

Les convolutions peuvent être efficacement calculées avec les transformations DCT en $O(N^2 \log N)$ opérations et pour une image couleur RGB, 3M convolutions doivent être calculées. La sommation sur n peut être parallélisée, puisque l'évaluation de $a_n(x) (\omega * I_n)(x)$ est indépendante pour chaque n , mais ça n'a pas été utilisé ici.

4.3 Implémentation pour la Correction des Images Sous-marines

4.3.1 Test de l'ACE sur des Images Synthétiques

La méthode ACE a été initialement implémentée à travers le programme `Undermarine_test2.py` qui utilise la fonction `ace_enhance_image_poly` pour des images synthétiques afin d'évaluer son efficacité dans des conditions idéales et d'observer les caractéristiques du filtrage. Certaines de ces propriétés dépendent des paramètres de l'algorithme, tandis que d'autres sont influencées par le contenu de l'image. À moins d'indication contraire, les résultats présentés dans ici utilisent :

- $\alpha = 7$.
- Un polynôme c de degré $D = 11$.
- Une distance euclidienne.

On remarque L'ACE ajuste automatiquement la luminosité et la dynamique de l'image en fonction de sa luminosité d'origine, contrairement à d'autres algorithmes qui augmentent systématiquement la luminosité. Par

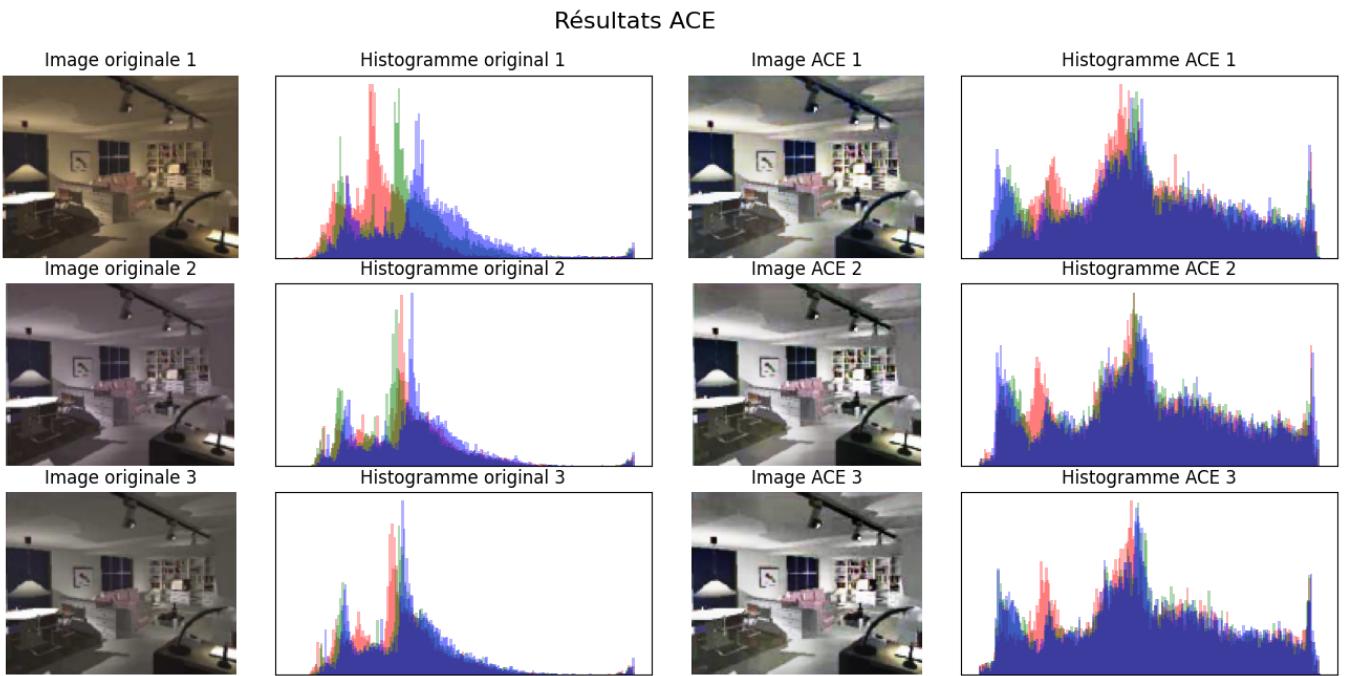


FIGURE 4.2 – Résultats de l'ACE sur des images synthétiques

exemple, il augmente la luminosité d'une image sous-exposée et la réduit pour une image surexposée. L'impact de l'algorithme sur la dynamique est visible dans les histogrammes avant et après filtrage ACE. Quantitativement, les effets sur la luminosité et la dynamique sont mesurés par :

- **Lissage de l'histogramme** : distance L^1 entre l'histogramme de l'image et un histogramme plat.
- **Pourcentage de dynamique utilisée** : proportion des valeurs dans l'étendue dynamique disponible.
- **Pourcentage de noir/blanc inutilisés** : fraction de valeurs inutilisées aux extrémités de l'étendue dynamique.

4.3.2 Correction des Images Sous-marines avec l'ACE

Nous appliquons maintenant la méthode aux images sous-marines utilisées précédemment. Cependant, nous rencontrons les mêmes problèmes observés plus tôt : pour les images dont la composition s'éloigne de l'hypothèse du monde gris, en particulier dans les zones maritimes avec de grandes étendues d'eau, l'algorithme tend à accentuer la teinte rouge, comme on le voit dans l'exemple du corail. Bien que les résultats obtenus avec l'espace $l\alpha\beta$ soient similaires à ceux des précédents essais, cette méthode est globalement plus performante. En effet, l'amélioration du contraste est meilleure, comme on peut le constater dans la FIGURE 4.3 (le reste des images traitées sont dans le fichier `Images_ACE_(alpha7-deg11-dist)`).

4.3.3 Application de l'ACE sur le Canal l

Une autre approche implémentée consiste à appliquer l'ACE uniquement sur le canal l après une transformation dans l'espace colorimétrique $l\alpha\beta$, dans le but d'améliorer le contraste et la luminosité de l'image. Cela est réalisé via la fonction `process_underwater_image`, avec des ajustements apportés à la fonction `gray_world_lab`, tout en supposant l'hypothèse du monde gris. Comme illustré dans la FIGURE 4.4 (le reste des images traitées sont dans le fichier `Images_Lab_GW_+_ACE(L)`), cette méthode permet une augmentation notable de la luminosité. Cependant, le contraste global en ressort réduit par rapport à l'application de l'ACE sur les trois canaux RGB. En outre, une différence importante réside dans la gestion des couleurs : contrairement à l'égalisation d'histogramme, qui équilibre les teintes de manière uniforme, l'ACE met fortement en avant la composante rouge de l'image.

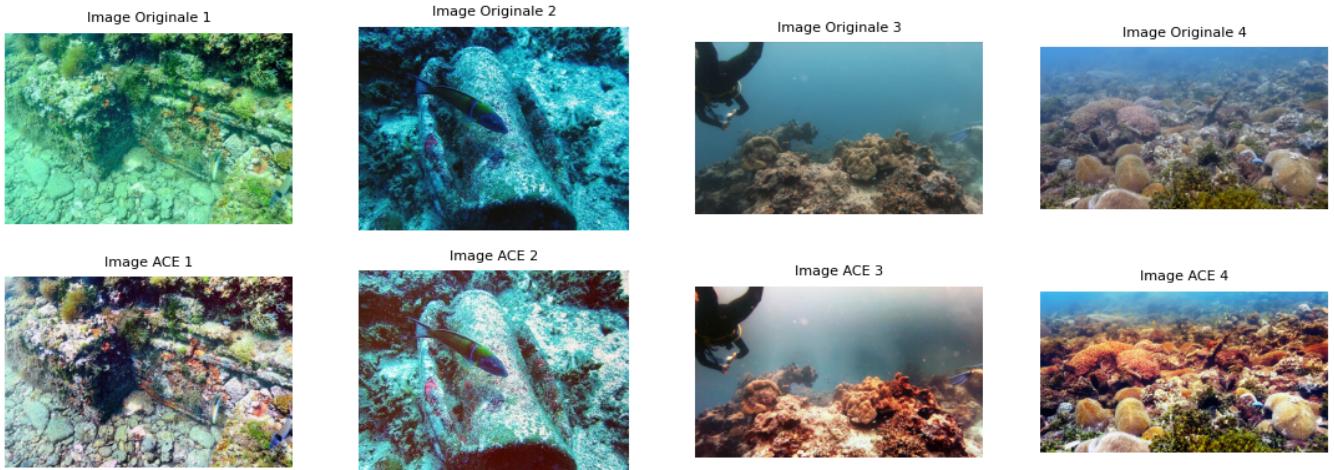
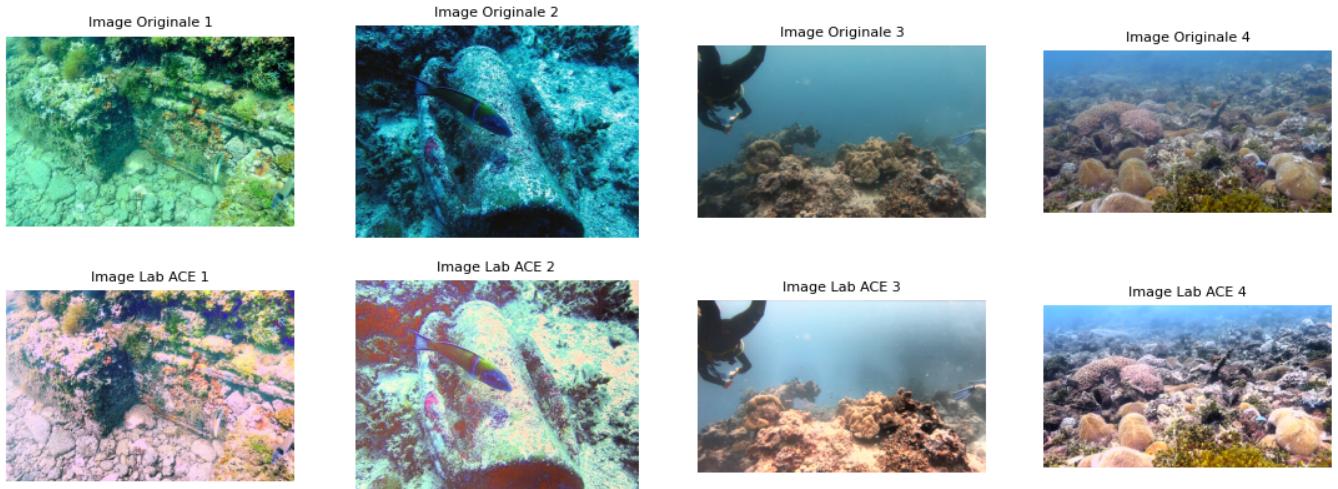


FIGURE 4.3 – Résultats de l'ACE sur les images sous-marines

FIGURE 4.4 – Résultats de l'ACE sur le canal l des images de la FIGURE 3.1

5 Conclusion

Comme nous avons pu le constater, la diversité des méthodes étudiées et expérimentées offre un large éventail de possibilités pour l'amélioration des images sous-marines. Chaque méthode possède des avantages. Toutefois, ces techniques présentent également des limites, notamment en termes de complexité computationnelle et d'adaptabilité à des conditions variées.

Lorsque l'on s'oriente vers des applications en temps réel, comme l'analyse de vidéos pour des robots sous-marins ou des systèmes de surveillance, la rapidité des algorithmes devient un critère essentiel. Les méthodes classiques, bien qu'efficaces dans des contextes spécifiques, peinent à répondre à cette exigence en raison du temps de traitement nécessaire. De ce fait, une approche moderne, basée sur l'exploitation de modèles d'apprentissage profond préalablement entraînés, s'impose comme une solution prometteuse. Ces modèles permettent de combiner **efficacité**, **rapidité** et **qualité**.

Table des figures

| | | |
|-----|--|----|
| 2.1 | Résultats de la correction RGB (avec l'hypothèse du monde gris) sur des images sous-marines | 5 |
| 2.2 | Résultats de la correction RGB (avec l'hypothèse des white patches avec $p = 2\%$) | 6 |
| 2.3 | Résultats de la correction RGB (avec l'hypothèse des white patches avec $p = 20\%$) | 6 |
| 3.1 | Résultats de la correction $l\alpha\beta$ (avec l'hypothèse du monde gris) sur des images sous-marines | 8 |
| 3.2 | Diagramme $\alpha\beta$ et Histogramme de la Luminance avant la correction | 9 |
| 3.3 | Diagramme $\alpha\beta$ et Histogramme de la Luminance après la correction | 9 |
| 3.4 | Résultats de l'égalisation d'histogramme sur le canal l des images de la FIGURE 3.1 | 10 |
| 4.1 | Fonction pente s_α | 11 |
| 4.2 | Résultats de l'ACE sur des images synthétiques | 13 |
| 4.3 | Résultats de l'ACE sur les images sous-marines | 14 |
| 4.4 | Résultats de l'ACE sur le canal l des images de la FIGURE 3.1 | 14 |

Références

- [1] R. Schettini and S. Corchs, "Underwater Image Processing : State of the Art of Restoration and Image Enhancement Methods," *EURASIP Journal on Advances in Signal Processing*, vol. 2010, Article ID 746052, 2010, doi : 10.1155/2010/746052.
- [2] G. Bianco, M. Muzzupappa, F. Bruno, R. Garcia, and L. Neumann, "A New Color Correction Method for Underwater Imaging," *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, vol. XL-5/W5, pp. 16–17, 2015, Underwater 3D Recording and Modeling, 16–17 April 2015, Piano di Sorrento, Italy.
- [3] E. Reinhard, M. Ashikhmin, B. Gooch, and P. Shirley, "Color Transfer between Images," *Applied Perception*, vol. 34, September/October 2001.
- [4] A. Rizzi, C. Gatta, and D. Marini, "A New Algorithm for Unsupervised Global and Local Color Correction," *Department of Information Technology, University of Milano*, 2001.
- [5] P. Getreuer, "Automatic Color Enhancement (ACE) and its Fast Implementation," *Image Processing On Line*, 2012, doi : 10.5201/ipol.2012.g-ace.
- [6] J. Han, M. Shoeiby, T. Malthus, E. Botha, J. Anstee, S. Anwar, R. Wei, M. A. Armin, H. Li, and L. Petersson, "Underwater Image Restoration via Contrastive Learning and a Real-World Dataset," *Remote Sensing*, vol. 14, no. 17, p. 4297, 2022. <https://doi.org/10.3390/rs14174297>.