



TÉLÉCOM PARIS

Projet IMA 201
Correction automatique de la couleur

Groupe :

Nourelhouda KLICH
Adnane EL BOUHALI

Encadrant :

Christophe KERVAZO

14 novembre 2023

Table des matières

1	Introduction	1
2	Fonctionnement de l'algorithme	2
2.1	L'ajustement spatial chromatique	2
2.1.1	Mise en œuvre de différentes fonctions $r(\cdot)$	2
2.1.2	Mise en œuvre de différentes distances $d(\cdot)$	3
2.2	Mise à l'échelle de la reproduction dynamique des tons	3
2.2.1	Mise à l'échelle linéaire	3
2.2.2	Mise à l'échelle Gray World White Patch	4
3	Analyse des résultats de l'algorithme ACE	5
3.1	ACE avec différentes fonctions $r(\cdot)$	5
3.2	ACE avec différentes fonctions $d(\cdot)$	7
3.3	Résultats génériques	9
3.4	Caractéristiques de filtrage	11
3.4.1	Constance de la luminosité et modification dynamique	11
3.4.2	Constance de la couleur	12
3.4.3	Modification de contraste	15
4	Conclusion	16

Chapitre 1

Introduction

Ce rapport présente notre implémentation d'un algorithme pour l'amélioration non supervisée des images numériques avec des mesures globales et effets locaux, appelés ACE pour Automatic Color Equalization.

Il est basé sur un modèle informatique de la vision humaine, système qui fusionne les deux mécanismes de péréquation globale de base "White Patch", qui suggère la présence d'une zone de référence blanche dans l'image, et « Gray World », qui implique que l'image peut être représentée par une valeur moyenne de gris.

ACE réalise un effet de filtrage local en prenant en compte l'espace couleur répartition dans l'image. Comme le système visuel humain, ACE est capable de s'adapter à des conditions d'éclairage très variables, et d'extraire efficacement les informations visuelles de l'environnement.

Il a montré des résultats prometteurs dans la réalisation de différentes tâches d'égalisation, par exemple assurer la constance des couleurs et de la luminosité et le contrôle du contraste. Dans ce rapport, nous détaillons les étapes clés de l'implémentation, les résultats obtenus, et une analyse détaillée de ces derniers.

Chapitre 2

Fonctionnement de l'algorithme

L'algorithme ACE repose sur deux mécanismes d'interaction spatiale qui contribuent au résultat final. En effet, le modèle est structuré en deux étapes ; la première étape gère l'ajustement spatial chromatique, en se concentrant sur la constance des couleurs et le réglage du contraste , et la deuxième étape permet de configurer la plage de sortie (output range) pour obtenir un mappage de tons précis et une constance de luminosité.

De plus, une condition préliminaire de l'algorithme ACE est la vérification des conditions de “Gray World / White Patch” . Dans ce contexte, pour l'espace colorimétrique RVB, chaque canal doit atteindre sa valeur maximale, soit 255. Ces deux hypothèses sont vérifiées pour toutes les images avant d'appliquer l'algorithme ACE.

2.1 L'ajustement spatial chromatique

Dans un premier temps, l'ajustement chromatique spatial produit une image de sortie R dans laquelle chaque pixel est calculé en fonction du contenu de l'image. Chaque pixel p de l'image de sortie R est calculé séparément pour chaque canal c comme suit :

$$Rc(p) = \frac{\sum_{j \in \text{Subset}, j \neq p} [r(Ic(p) - Ic(j))/d(p, j)]}{\sum_{j \in \text{Subset}, j \neq p} [rmax/d(p, j)]} \quad (2.1)$$

Où c est le canal de couleur, $I_c(p) - I_c(j)$ rend compte du mécanisme d'inhibition latérale, $d(\cdot)$ est une fonction de distance qui pondère l'importance de la contribution locale ou globale, $r(\cdot)$ est la fonction discutée ci-dessous qui rend compte de l'apparence de luminosité relative du pixel, $rmax$ est la valeur maximale de cette dernière et le Subset est un sous-ensemble de pixels centré autour du pixel p (ou l'image entière).

2.1.1 Mise en œuvre de différentes fonctions $r(\cdot)$

Le rôle de la fonction $r(\cdot)$ est de quantifier l'influence de chaque pixel de l'image d'entrée sur chaque pixel de l'image de sortie.

Afin de respecter nos deux hypothèses de base (White Patch et Gray World), notre fonction $r(\cdot)$ est soumise à certaines contraintes :

1. Imparité de $r(\cdot)$: Affirmer le comportement du Grey World.
2. Non-linéarité de $r(\cdot)$: Cette contrainte est assurée par la saturation de $r(\cdot)$.

Les différentes fonctions $r(\cdot)$ utilisées sont les suivantes :

$$\text{Linéaire : } r(x) = x$$

$$\text{Signum : } r(x) = \text{sign}(x)$$

$$\text{Saturation : } r(x) = \min(1.0, \max\left(\frac{x}{a}, -1.0\right)) \quad (\text{avec } a \text{ le slope de la fonction})$$

Pour sélectionner la fonction $r(\cdot)$ la plus appropriée, nous allons tester chacune d'entre elles sur une image avec une fonction de distance fixe et nous discuterons les résultats.

2.1.2 Mise en œuvre de différentes distances $d(\cdot)$

Pour mettre en œuvre l'ACE, nous devons tenir compte de la distribution spatiale des couleurs dans l'image. Pour ce faire, nous devons calculer la distance entre chaque pixel et les autres pixels du subset. Pour cela, nous définissons les différentes distances suivantes :

$$\text{Euclidienne : } \sqrt{dx^2 + dy^2}$$

$$\text{Exponentielle inverse : } \frac{1}{\exp\left(-\frac{\alpha}{Ed}\right)} \quad (\text{avec } Ed \text{ la norme euclidienne})$$

$$\text{Manhattan : } dx + dy$$

$$\text{Maximum : } \max(dx, dy)$$

Comme on l'a fait pour la fonction $r(\cdot)$, pour choisir la distance qui fonctionne le mieux, nous testons l'algorithme avec différentes distances et une fonction $r(\cdot)$ fixe. Ce point sera abordé dans la section sur l'analyse des résultats.

2.2 Mise à l'échelle de la reproduction dynamique des tons

Cette étape permet de mapper les pixels de l'image intermédiaire R_c obtenue par la première étape dans l'image de sortie O_c . Pour le faire, on peut utiliser deux approches différentes, la première est la réalisation d'une simple maximisation dynamique (mise à l'échelle linéaire) et la deuxième consiste à prendre en compte les conditions "Gray World" et "White Patch" et de cartographier les valeurs d'apparence de luminosité relative de chaque canal.

2.2.1 Mise à l'échelle linéaire

Cette méthode est simple, elle met à l'échelle linéairement la plage des valeurs dans l'image R indépendamment dans le canal relatif c dans la plage $[0, 255]$, selon cette formule :

$$O_c(p) = \text{round}\left(\frac{s_c(R_c(p)))}{m_c}\right)$$

avec :

$$s_c : \text{le slope du segment } [(m_c, 0), (M_C, 255)]$$

M_c : la valeur maximale de pixel dans R_c

m_c : la valeur minimale de pixel dans R_c

Cette méthode était implémentée dans notre code à travers la fonction `linear_scaling_0c(Rc)`.

2.2.2 Mise à l'échelle Gray World White Patch

Cette deuxième méthode est une approche améliorée qui met à l'échelle linéairement les valeurs de l'image R en utilisant la formule suivante :

$$O_c(p) = \text{round}(127.5 + sc \cdot Rc(p))$$

En considérant M_c comme la référence blanche et la valeur zéro dans R_c comme une approximation du point moyen de référence grise pour calculer le slope s_c .

Cette méthode était implémentée dans notre code à travers la fonction `whiteGray_0c(Rc)`

Chapitre 3

Analyse des résultats de l'algorithme ACE

3.1 ACE avec différentes fonctions $r(\cdot)$

Nous avons testé différentes fonctions $r(\cdot)$ avec la distance euclidienne pour trouver le meilleur compromis entre la correction de la couleur et du contraste. Les figures suivantes représentent les différents tests réalisés sur la même image :

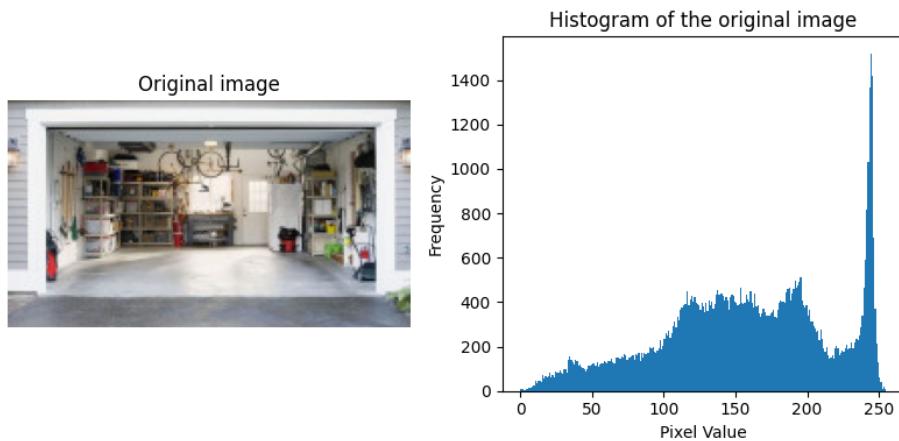


FIGURE 3.1 – Image originale

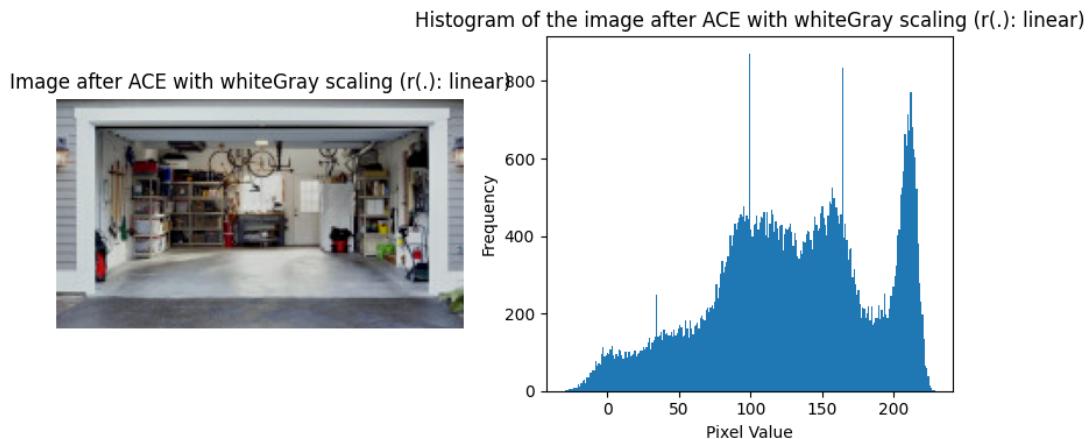


FIGURE 3.2 – Linéaire

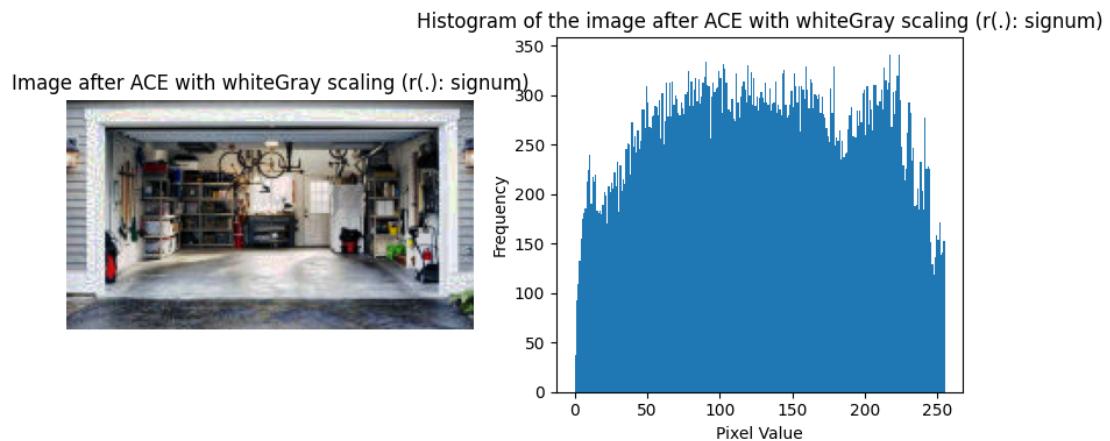


FIGURE 3.3 – Signum

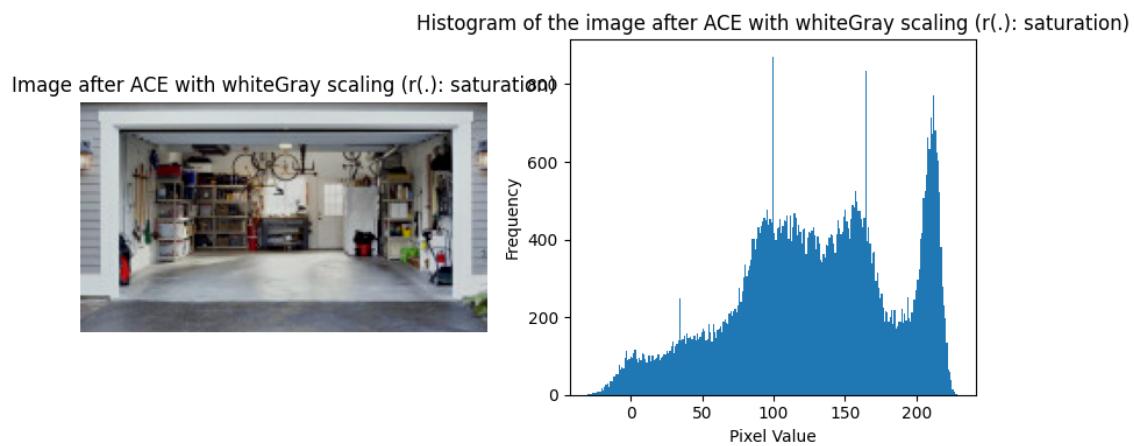


FIGURE 3.4 – Saturation

On remarque que la variation de la pente de $r(.)$ agit comme un accordeur de contraste. En effet, plus la pente est importante, plus le contraste est élevé.

Finalement, on choisit la fonction signum, vu qu'elle présente les meilleurs résultats. En effet, en analysant les images et leurs histogrammes, on remarque que les résultats de la fonction signum sont considérablement plus contrastés, on peut également constater une correction de la couleur dans toute l'image et que l'histogramme est plus aplati, couvrant presque toute la gamme des valeurs possibles.

3.2 ACE avec différentes fonctions $d(\cdot)$

Dans cette partie, on a testé l'algorithme ACE plusieurs fois sur la même image en fixant la fonction $r(\cdot)$ (à Signum) et en modifiant à chaque fois la fonction $d(\cdot)$ (euclidienne, inverse exponentiel, manhattan, maximum). Et cela afin d'identifier la fonction distance qui donne les meilleurs résultats visuellement. Les figures suivantes montrent les images et leurs histogrammes respectifs après application de ACE avec une fonction distance différente sur chacune :

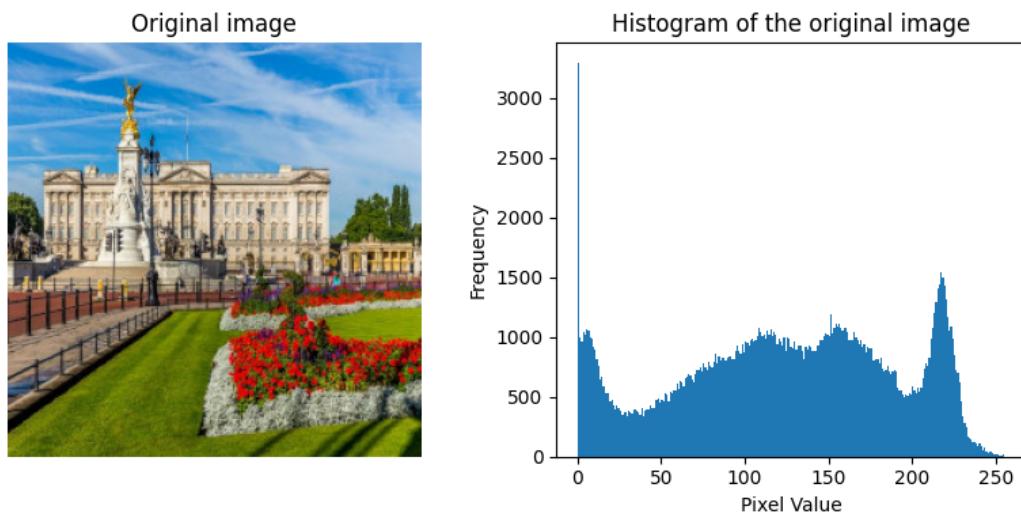


FIGURE 3.5 – Image originale

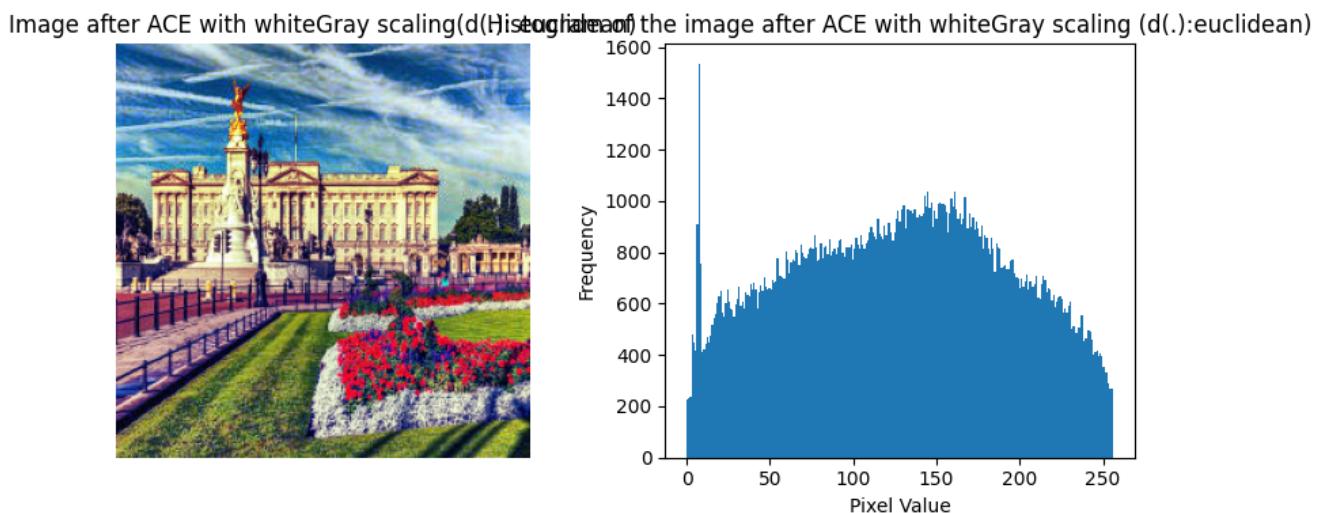


FIGURE 3.6 – Euclidienne

Image after ACE with whiteGray scaling(d(Histogramme_exponentiel) after ACE with whiteGray scaling (d(.):inverse_exponential)

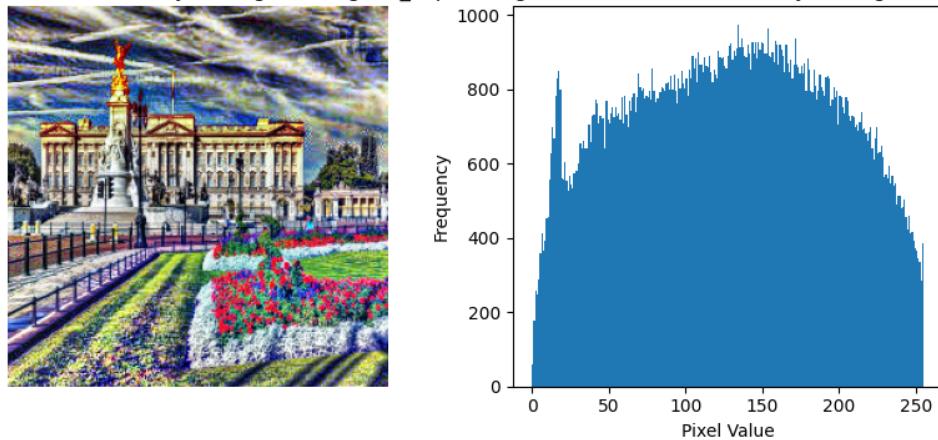


FIGURE 3.7 – Inverse exponentielle

Image after ACE with whiteGray scaling(d(Histogramme_manhattan) after ACE with whiteGray scaling (d(.):manhattan)

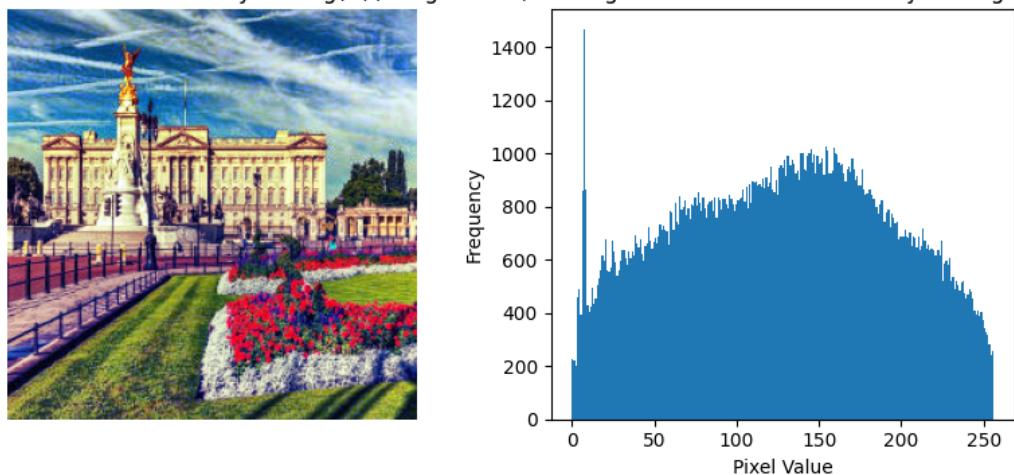


FIGURE 3.8 – Manhattan

Image after ACE with whiteGray scaling(d(Histmax)of the image after ACE with whiteGray scaling (d(.):maximum)

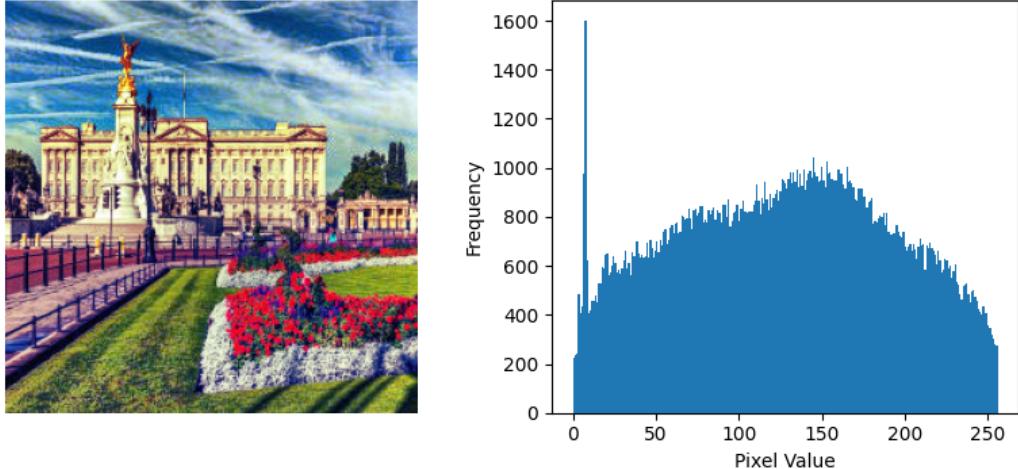


FIGURE 3.9 – Maximum

On observe que les résultats pour les distances euclidienne, maximum et manhattan sont proches mais il semble que la fonction de distance euclidienne donne des résultats un peu meilleures que les distances maximum et de Manhattan, avec un bon équilibre entre le contraste et la préservation des détails. Tandis que la fonction de distance exponentielle inverse a un produit certaine déformation de couleurs et des détails sur l'image.

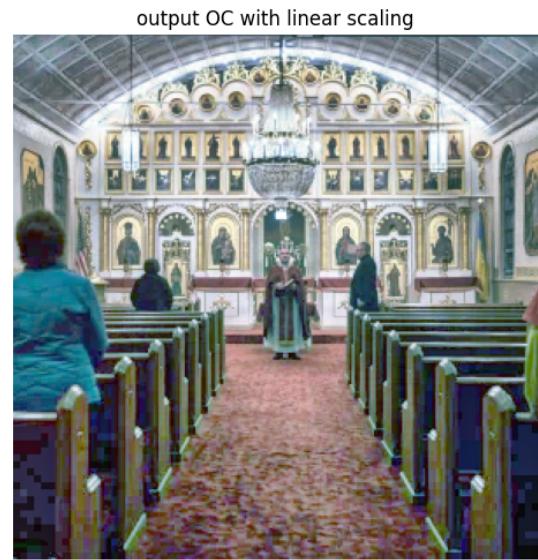
Après avoir appliqué ce test sur différentes images, on a décidé d'utiliser la distance euclidienne pour tester notre algorithme ACE, ce qui confirme l'hypothèse de l'article de recherche qui ont choisi la distance euclidienne pour appliquer ACE.

3.3 Résultats génériques

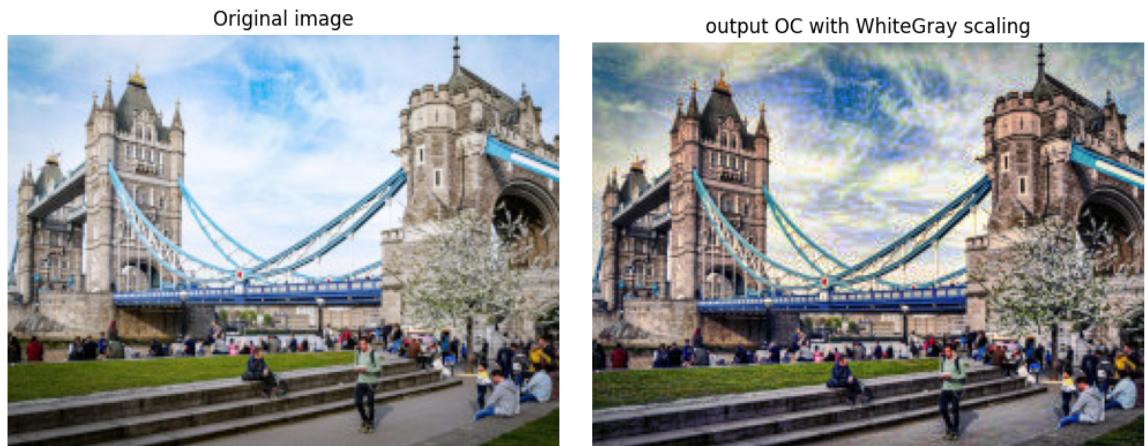
Notre implémentation de l'algorithme ACE a donné des résultats globalement satisfaisants. L'algorithme a réussi à améliorer significativement la couleur des images, rendant les détails plus visibles et améliorant la perception visuelle.



(a) Image originale



(b) ACE



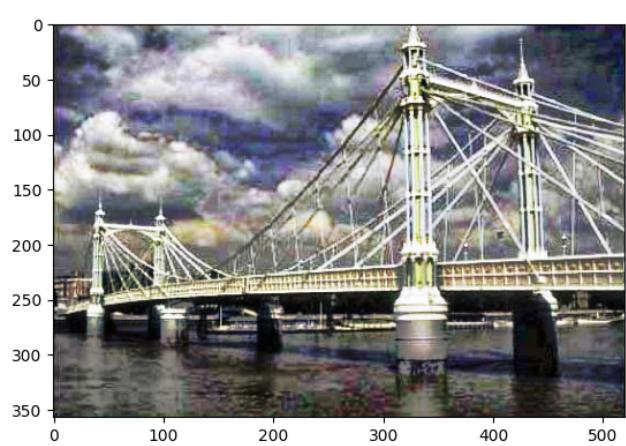
(c) Image originale



(d) ACE



(e) Image originale



(f) ACE

FIGURE 3.10 – Différents résultats de notre implémentation

3.4 Caractéristiques de filtrage

Dans cette partie, on étudie des caractéristiques plus significatives de l'algorithme ACE.

En se référant à l'article de recherche, on a utilisé les paramètres suivants pour appliquer ACE :

- La fonction de saturation avec slope 20 pour $r(\cdot)$
- La fonction euclidienne pour $d(\cdot)$
- La mise à l'échelle White Patch/Gray World

Remarque : On a adapté notre algorithme afin de pouvoir illustrer ces caractéristiques sur des images noir et blanc.

3.4.1 Constance de la luminosité et modification dynamique

“L’intégration du principe Grey World par ACE lui permet d’ajuster la luminosité selon sa valeur originale” [1] .

Les figures suivantes illustrent cette proposition, en appliquant ACE sur des images en différentes conditions d’exposition.

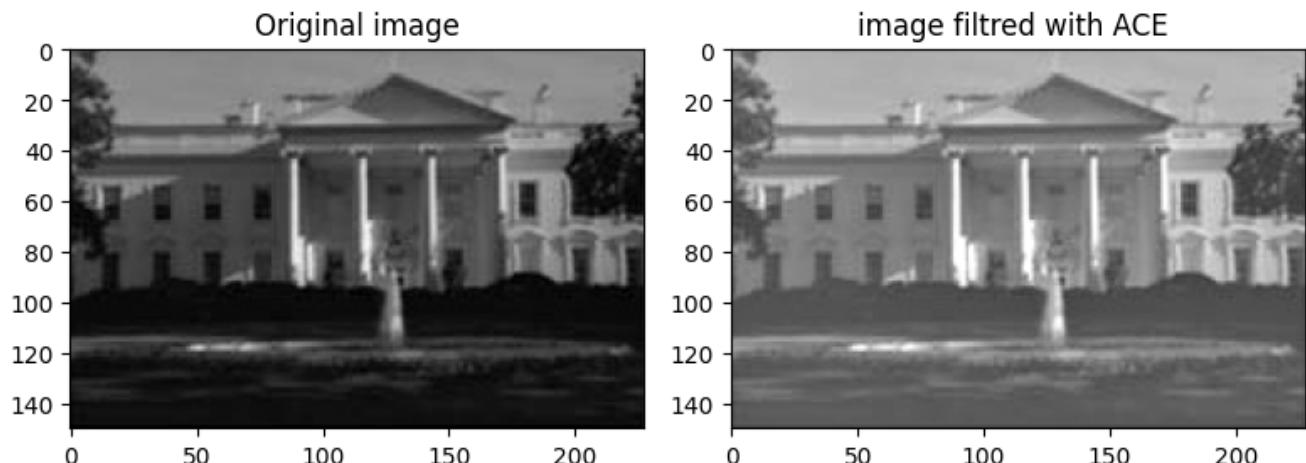


FIGURE 3.11 – Image originale (gauche) et ACE (droite) (1)

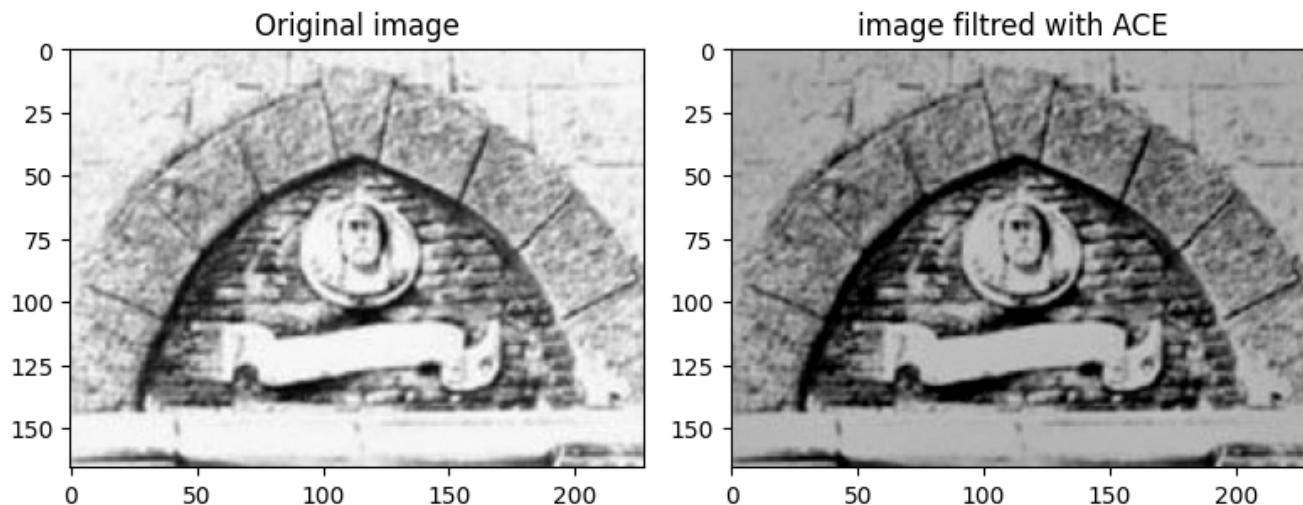


FIGURE 3.12 – Image originale (gauche) et ACE (droite) (2)

On peut observer sur la première figure que pour l'image 1 qui est légèrement sous-exposée, l'algorithme ACE augmente la luminosité moyenne de l'image, tandis que pour l'image 2 qui est surexposée, ACE permet de diminuer la luminosité.

De plus, on peut observer l'effet de ACE sur la dynamique de l'image sur la deuxième figure ci-dessus, où l'image originale semble avoir une gamme dynamique autour des valeurs spécifiques. Et l'image après ACE montre des changements visibles sur sa dynamique.

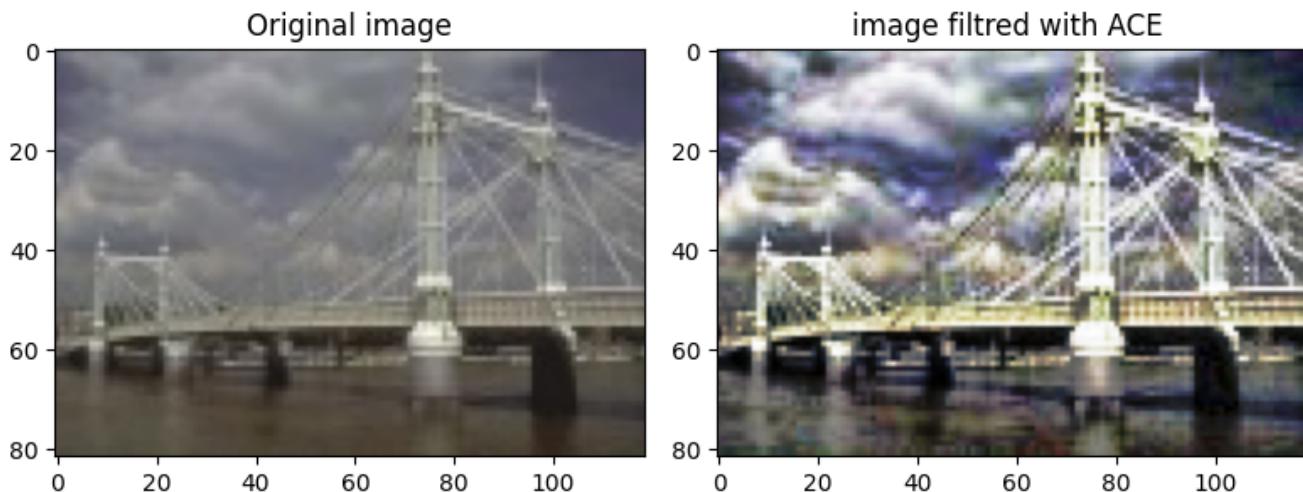


FIGURE 3.13 – Plage dynamique d'une image avant et après ACE

3.4.2 Constance de la couleur

Afin d'évaluer la capacité de la constance de couleur pour l'algorithme ACE, on a utilisé un ensemble d'images synthétiques (générées à l'aide de Gimp en modifiant les conditions d'éclairage). En appliquant ACE sur ces images, on peut observer comment l'algorithme effectue l'ajustement de couleur aux différents types d'éclairage synthétique. En plus de l'observation visuelle, on utilise

la métrique DELTA E moyenne qui représente une mesure de la différence de couleurs perçue entre deux images et donnée par la formule :

$$\Delta E_{\text{mean}} = \frac{1}{\text{size}_x \times \text{size}_y} \sum_{p \in \text{Image}} \Delta E(I_1(p), I_2(p))$$

En se basant sur les figures suivantes, une évaluation visuelle du filtrage ACE sur les images synthétiques montre une réduction importante de la dominante chromatique des illuminants pour chacune des images A, B et C ainsi qu'une forte ressemblance entre les résultats du filtrage ACE pour ces images.

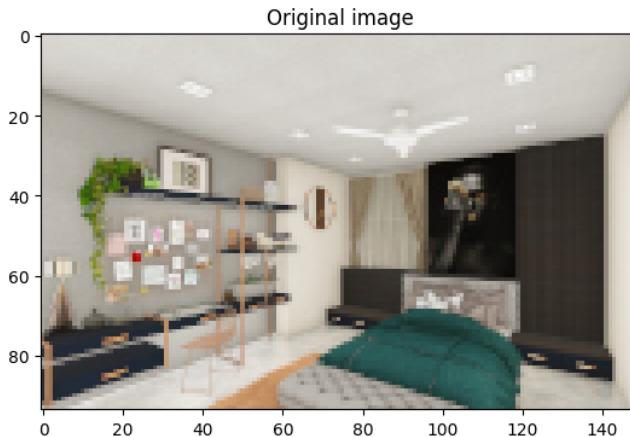


FIGURE 3.14 – Image originale

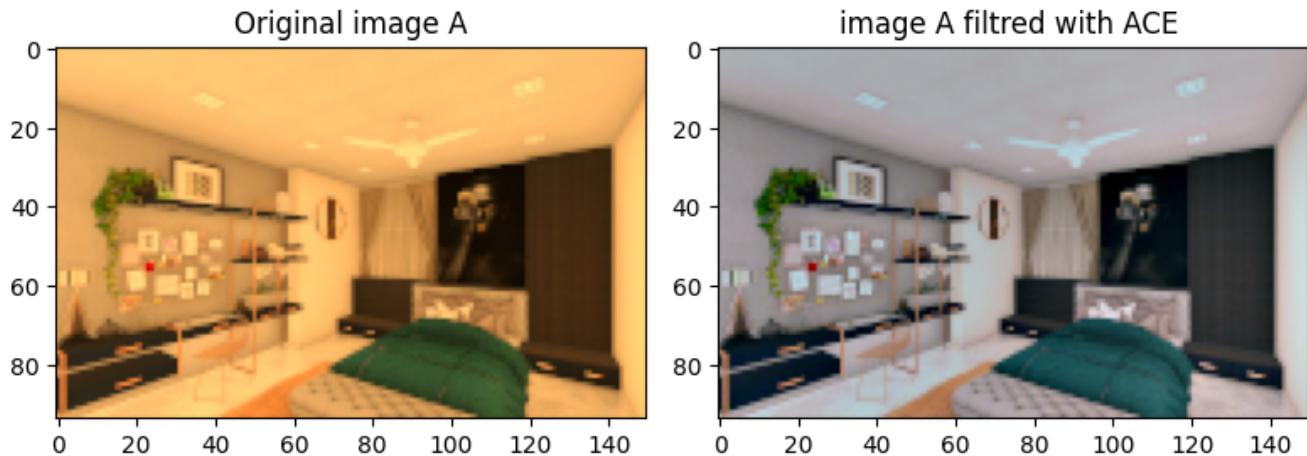


FIGURE 3.15 – Comparaison visuelle du filtrage ACE sur les images synthétiques

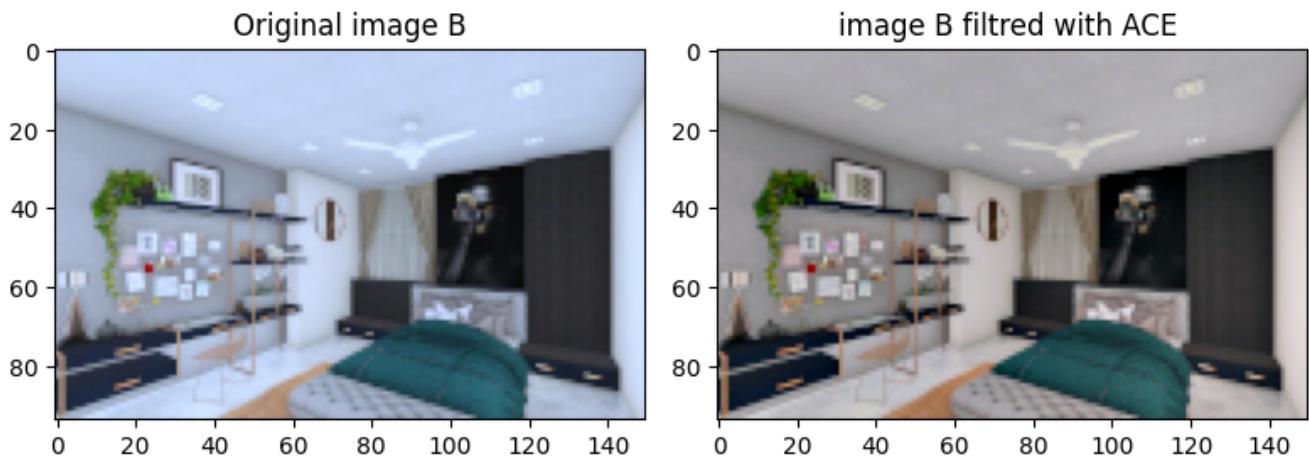


FIGURE 3.16 – Comparaison visuelle du filtrage ACE sur les images synthétiques

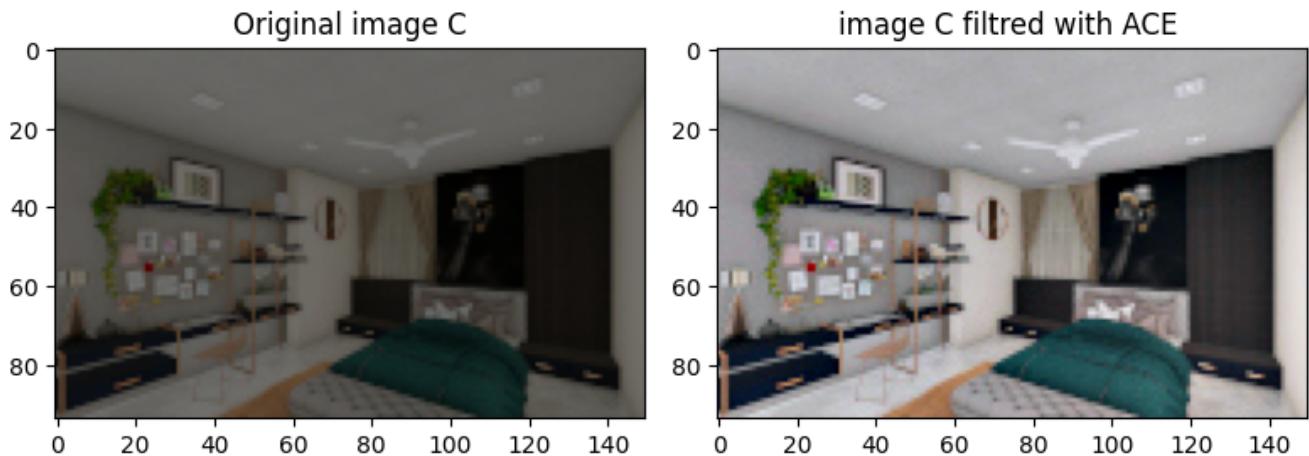


FIGURE 3.17 – Comparaison visuelle du filtrage ACE sur les images synthétiques

Et les figures ci-dessous, présentent les valeurs de ΔE pour chaque couple d’image avant et après le filtrage ACE, confirme notre observation visuelle. En effet, comme on peut voir sur les deux tableaux, la distance chromatique moyenne mesurée entre chaque paire d’images diminue après le filtrage par ACE. Ce qui montre que ACE permet de maintenir des couleurs cohérentes malgré les différentes variations d’éclairage.

Mean DE on the original synthetic images			
	Image A	B	C
A	0.0	105.94	136.21
B	105.94	0.0	147.23
C	136.21	147.23	0.0

FIGURE 3.18 – Images avant ACE

Mean DEon the ACE filtered synthetic images			
	Image A	B	C
A	0.0	4.52	3.97
B	4.52	0.0	0.81
C	3.97	0.81	0.0

FIGURE 3.19 – Images après ACE

3.4.3 Modification de contraste

L'impact de modification de contracte pour l'algorithme ACE est caractérisé par sa capacité à affecter à la fois le contraste global et le contraste local d'une image. On peut observer sur la figure suivante que le contraste global de l'image diminue, alors que le contraste local augmente (on peut mieux voir les petits détails de la voiture).

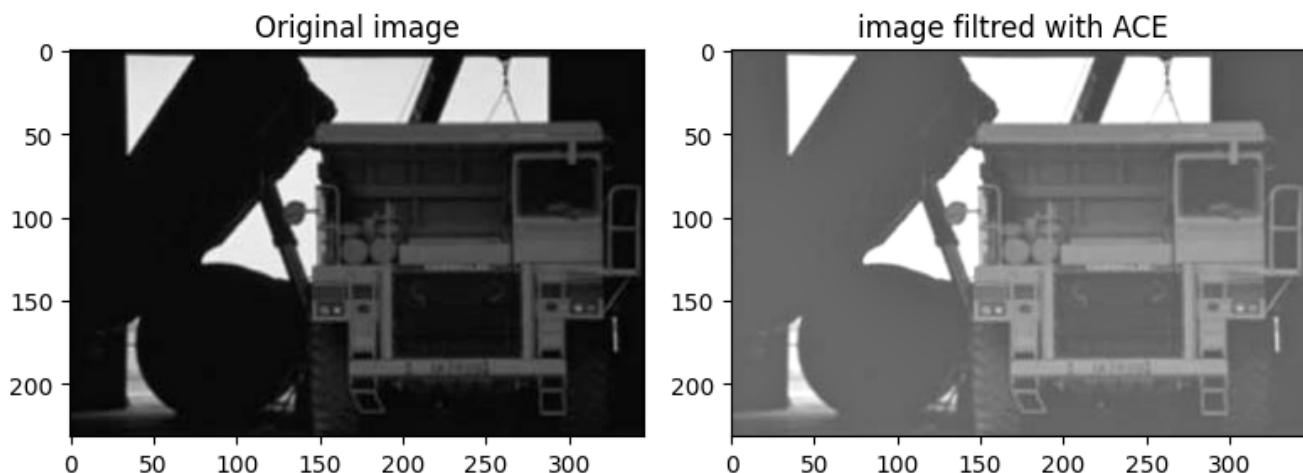


FIGURE 3.20 – Effet de modification de contraste

Chapitre 4

Conclusion

On peut conclure à partir des résultats obtenus que l'algorithme ACE est extrêmement utile pour la correction des couleurs. En effet, ACE a démontré son efficacité à réaliser une bonne correction de la constance des couleurs et une égalisation des tons satisfaisante en effectuant une correction de l'image locale et globale. Par contre, la complexité spatiale ainsi que la complexité temporelle de l'algorithme sont très élevées. On a tenté d'optimiser cela en utilisant le minimum possible de boucles dans l'algorithme ainsi qu'en considérant des sous-ensemble de l'image, cette approche a permis d'optimiser la complexité. Mais, certaines autres améliorations peuvent être envisagées pour ce problème, comme l'approximation polynomiale qui consiste à effectuer une estimation polynomiale pour la fonction de pente (slope).

Bibliographie

- [1] Daniele Marini Alessandro Rizzi, Carlo Gatta. A new algorithm for unsupervised global and local color correction. 2002.
- [2] Pascal Getreuer. A new algorithm for unsupervised global and local color correction. *IOPL Journal*, 2, 2012.