

RAPPORT

Projet Vidéo Numérique

COURDY-BAHSOUN CLÉMENCE
M2 IGAI

15.02.2020

Projet

1 Introduction

L'objectif de ce projet est de mener une étude comparative des différentes méthodes d'appariement de bloc pour faire de la reconstruction d'image, ou de la mise en évidence du mouvement.

Pour ce faire nous avons du prendre des images successives dans des vidéos afin de tester les différentes méthodes imposées par le sujet.

Pour ce faire nous avons utilisé le logiciel en ligne kapwing pour extraire 5 couples d'images sur 3 vidéos différentes, afin d'illustrer les résultats obtenus pour différentes sortes de mouvement.



FIGURE 1 – image 1, séquence 1, vidéo : teaser de kaamelott



FIGURE 2 – image 2, séquence 1, vidéo : teaser de kaamelott

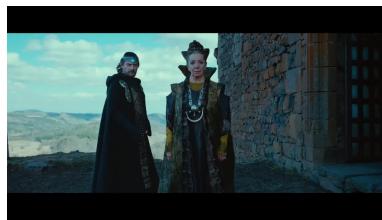


FIGURE 3 – image 1, séquence 2, vidéo : teaser de kaamelott

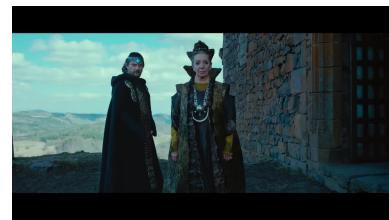


FIGURE 4 – image 2, séquence 2, vidéo : teaser de kaamelott



FIGURE 5 – image 1, séquence 3, vidéo : teaser de kaamelott



FIGURE 6 – image 2, séquence 3, vidéo : teaser de kaamelott



FIGURE 7 – image 1, vidéo : générique de friends



FIGURE 8 – image 2, vidéo : générique de friends



FIGURE 9 – image 1, vidéo : générique de westworld



FIGURE 10 – image 2, vidéo : générique de westworld

2 Cas 1 : Recherche exhaustive

2.1 Principe

L'objectif de cette méthode est de reconstruire l'image 2 par bloc de voisinage 8, soit des blocs 3x3. Pour chaque bloc de l'image deux on cherche le bloc dans un voisinage de 16x16 de l'image 1 qui aura la plus petite erreur quadratique moyenne. Ce bloc de l'image 1 deviendra le bloc de l'image reconstruite.

Ainsi l'appel à la fonction **reconstruction(im1,im2,window,N)** avec window = 16 et N = 1 permet de reconstruire l'image 2 à partir de l'image avec un voisinage de parcours de 16x16 en calculant le meilleur bloc de manière exhaustive grâce à la fonction **calculerMeilleurBloc1(windowIm1,bloc2)** qui retourne le meilleure bloc en calculant l'erreur quadratique moyenne à partir de la fonction **eqm(bloc1,bloc2)**.

2.2 Résultats

L'image 11 est la reconstruction obtenue pour les images de la séquence 1 de kaamelott 1. L'erreur quadratique moyenne entre l'image 2 et l'image reconstruite est de 17,59. Ici seul les personnages se déplace en marchant. La caméra ne bouge pas, le mouvement est donc au niveau des ombres et également visible par la taille

du soleil plus ou moins obstrué en fonction de l'image. La reconstruction au niveau du paysage où il n'y a pas de mouvement est parfaite. Cependant au niveau des personnages on est encore trop près de l'image 1 bien que les bords des ombres ne soient plus lisses.



FIGURE 11 – Reconstruction de l'image 2 à partir de l'image 1 de la séquence 1 de kaamelott, eqm = 17,59

L'image 12 est la reconstruction obtenue pour les images de la séquence 2 de kaamelott 1. L'erreur quadratique moyenne entre l'image 2 et l'image reconstruite est d'environ 7,19. On remarque que le mouvement entre les deux images étant infime (au niveau des cheveux de l'homme, et de la robe de la femme du au vent) notre résultat est très proche de celui attendu, comme en témoigne l'errreur quadratique moyenne.

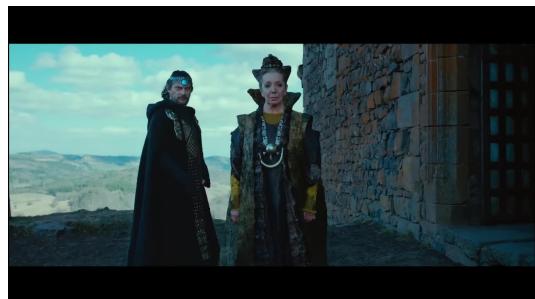


FIGURE 12 – Reconstruction de l'image 2 à partir de l'image 1 de la séquence 2 de kaamelott, eqm = 7,19

L'image 13 est la reconstruction obtenue pour les images de la séquence 1 de kaamelott 1. L'erreur quadratique moyenne entre l'image 2 et l'image reconstruite est de 22,02. Ici les personnages se déplacent en marchant. La caméra se déplace

légèrement aussi en suivant les acteurs. Aussi on remarque du flou lié à la reconstruction au niveau du mouvement des acteurs et sur certaines partie du désert.



FIGURE 13 – Reconstruction de l'image 2 à partir de l'image 1 de la séquence 3 de kaamelott, eqm = 22,02

L'image 14 est la reconstruction obtenue pour les images tirées du générique de friends 1. L'erreur quadratique moyenne entre l'image 2 et l'image reconstruite est de 54,44. Cette scène représente des jeunes gens dansant dans une fontaine. Les deux images successives présentent donc beaucoup de différences car il s'agit d'une scène avec de nombreux mouvements amples. De plus la fontaine met également des molécules d'eau en mouvement. Il est donc très difficile de reconstruire une image proche de celle attendue : les visages sont très floutés, les contours également, et pour l'individu au premier plan on voit que le bras a mal été reconstruit, il reste au milieu du jet d'eau.



FIGURE 14 – Reconstruction de l'image 2 à partir de l'image 1 de friends, eqm = 54,44

L'image 15 est la reconstruction obtenue pour les images du générique de west-world 1. L'erreur quadratique moyenne entre l'image 2 et l'image reconstruite est

seulement de 4,25. Ceci est directement lié au mouvement qui est représenté entre les deux images : seul les pattes arrières du cheval bouge, tout le reste de la scène est statique, ainsi on peut voir une légère pixélisation au niveau des pattes arrières, mais le résultat est très proche de celui attendu.



FIGURE 15 – Reconstruction de l'image 2 à partir de l'image 1 de westworld, eqm = 4,25

3 Cas 2 : Recherche en log2

3.1 Principe

La recherche en log consiste à rechercher les blocs de l'image 1 dans un voisinage initial avec l'erreur quadratique moyenne la plus petite possible sur les blocs voisins en croix (le bloc du dessus, celui du dessous et ceux de droite et de gauche). Pour ce faire on exécute la fonction **reconstruction(im1,im2,window,N)** avec window = 16 et N = 2 pour lancer la reconstruction avec **calculMeilleurBloc2(window,block)** qui effectue la recherche en log2.

3.2 Résultats

L'image 16 est la reconstruction obtenue pour les images de la séquence 1 de kaamelott 1. L'erreur quadratique moyenne entre l'image 2 et l'image reconstruite est de 17,80. D'un point de vue visuel on remarque tout de même que les bords des personnages qui marchent sont moins lisses que le résultat obtenu avec la première méthode 11



FIGURE 16 – Reconstruction Log2 de l'image 2 à partir de l'image 1 de la séquence 1 de kaamelott, eqm = 17,80

L'image 17 est la reconstruction obtenue pour les images de la séquence 2 de kaamelott 1. L'erreur quadratique moyenne entre l'image 2 et l'image reconstruite est de 7,74. On remarque que les détails sur les visages ne sont pas net, il semble avoir été lissé, et sont donc moins précis que ce qui est attendu.

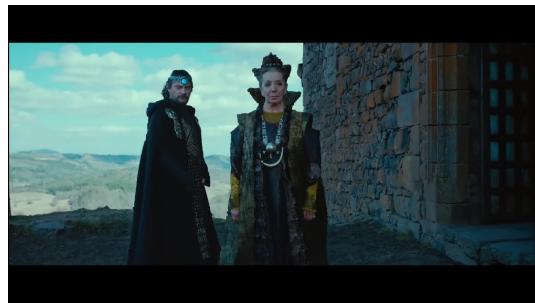


FIGURE 17 – Reconstruction Log2 de l'image 2 à partir de l'image 1 de la séquence 2 de kaamelott, eqm = 7,74

L'image 18 est la reconstruction obtenue pour les images de la séquence 3 de kaamelott 1. L'erreur quadratique moyenne entre l'image 2 et l'image reconstruite est de 22,91. On constate également un résultat de moins bonne qualité surtout au niveau des détails, c



FIGURE 18 – Reconstruction Log2 de l'image 2 à partir de l'image 1 de la séquence 3 de kaamelott, eqm = 22,91

L'image 19 est la reconstruction obtenue pour les images du générique de friends 1. L'erreur quadratique moyenne entre l'image 2 et l'image reconstruite est de 56,83. Ici l'erreur quadratique moyenne est plus élevée que pour la version d'appariement par recherche exhaustive 14. Cependant d'un point de vue visuel les deux images de reconstructions semblent d'aussi mauvaise qualité l'une que l'autre.



FIGURE 19 – Reconstruction Log2 de l'image 2 à partir de l'image 1 du générique de friends, eqm = 56,83

L'image 20 est la reconstruction obtenue pour les images du générique de westworld 1. L'erreur quadratique moyenne entre l'image 2 et l'image reconstruite est de 4,83. Ici encore, le résultat de l'erreur quadratique moyenne n'est pas beaucoup plus élevé que pour la version par recherche exhaustive, cependant le résultat est beaucoup moins précis visuellement, on remarque notamment que les détails fin ne sont pas conservés y compris sur les zones qui n'ont pas subit de déplacement.



FIGURE 20 – Reconstruction Log2 de l'image 2 à partir de l'image 1 du générique de westworld, eqm = 4,83

4 Cas 4 : Lukas et Kanade

4.1 Principe

Lukas et Kanade est une méthode qui permet de définir les vecteurs de déplacement entre deux images consécutives d'une scène. Ici on calcul le vecteur de déplacement sur le pixel central d'un bloc de 8 voisins, puis on passe au bloc suivant en décalant la fenêtre de 8 pixels (en ligne et/ou colonne) pour limiter le temps d'exécution. Une fois l'intégralité des vecteurs de déplacement pour tous les blocs de l'image calculé, nous reconstruisons l'image en faisant une translation du vecteur de déplacement pour trouver l'indice central du bloc de l'image 2 à utiliser pour reconstruire l'image 1.

Pour ce qui est du calcul des vecteurs de déplacement, la version faite en TP pour des images bitmap a été reprise et adapté pour des images RGB. Pour ce faire nous sommes les trois composantes pour le calcul de la matrice H et b nécessaire à la résolution du système linéaire pour le calcul des vecteurs de déplacement.

Deux versions de ont été testé : la première reconstruit l'image en partant sur un fond d'image noire, la seconde la reconstruit sur l'image attendue.

Pour lancer la reconstruction on exécute **casLK(im1,im2)**. Cette fonction calcule la matrice des vecteurs de déplacements en appelant la fonction **lukasKanade(im1,im2)**, puis on reconstruit l'image avec **calculerIm(im1,im2,deltaIm)**.

4.2 Résultats

La version 1 donne de très mauvais résultat, dut au fait qu'on applique pas le calcul du vecteur de mouvement à tous les pixel mais seulement au pixel central des blocs de 8, de l'image avec un décalage de 8 entre chaque pixel. Ainsi l'erreur quadratique moyenne est très élevé, et l'image n'est pas bien reconstruite.

La version 2 présente des erreurs quadratique moyenne très basses car la reconstruction se fait par dessus l'image 1. Ainsi elle nous permet de mettre en évidence le mouvement entre l'image 1 et l'image 2 sur les zones ayant subit un déplacement. Néanmoins il ne s'agit pas d'une reconstruction précise.



FIGURE 21 – Lukas et kanade v1 séquence 1 de kaamelott, eqm = 60,39



FIGURE 22 – Lukas et kanade v2 séquence 1 de kaamelott, eqm = 1,97

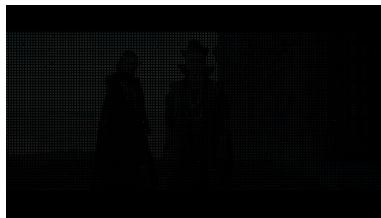


FIGURE 23 – Lukas et kanade v1 séquence 2 de kaamelott, eqm = 85,14

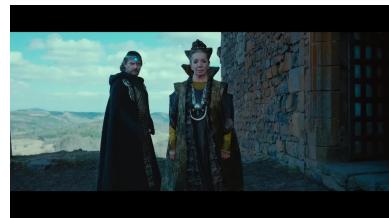


FIGURE 24 – Lukas et kanade v2 séquence 2 de kaamelott, eqm = 1,04



FIGURE 25 – Lukas et kanade v1 séquence 3 de kaamelott, eqm = 79,19



FIGURE 26 – Lukas et kanade v2 séquence 3 de kaamelott, eqm = 2,7

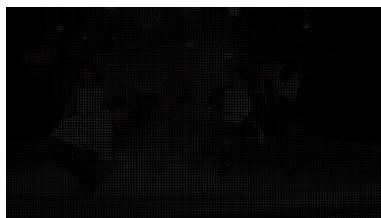


FIGURE 27 – Lukas et kanade v1 générique friends, eqm = 99,53



FIGURE 28 – Lukas et kanade v2 générique de friends, eqm = 5,5

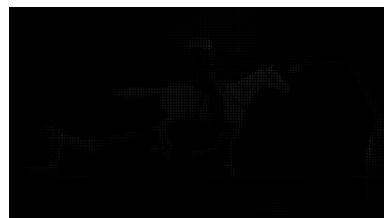


FIGURE 29 – Lukas et kanade v1 générique de westworld, eqm = 80,35



FIGURE 30 – Lukas et kanade v2 générique de westworld, eqm = 0,55

5 Conclusion

Les résultats visuels et ceux du calcul de l'erreur quadratique moyenne nous montre que sur les 3 cas testés (recherche exhaustive, recherche Log2 et Lukas & Kanade) les résultats les plus précis sont obtenus par la recherche exhaustive. Ensuite on obtient des résultats corrects avec le log2, en revanche Lukas et kanade donne de très mauvais résultat, en raison du fait qu'on ne calcul pas le vecteur de déplacement de tous les pixels mais seulement des blocs de 8.

On remarque également que plus le mouvement est important et plus il est difficile à reconstruire une image, d'autant plus quand l'environnement est également en mouvement.

De plus on constate que la reconstruction n'est tout de même pas assez proche du résultat attendu. Ceci est du à la taille très petite du voisinage utilisé pour la recherche exhaustive et la recherche en log2. Pour avoir de meilleur résultat il faudrait travailler sur des fenêtres de voisinage beaucoup plus grande, cependant le temps de calcul serait lui aussi beaucoup plus conséquent.

Je signale que je n'ai pas pu traiter les cas 3 et 5 en raison du manque de temps, le mois de février devant être consacré au chef d'œuvre il m'est difficile de terminer le projet sans que cela soit en défaveur du travail que je dois réaliser pour le chef d'œuvre.