VISION

Rapport

Disparity Map Estimation Using Graph Cuts

Étudiant :
Robin SOARES

Supervisé par :
Pascal MONASSE

Repository GitHub : `https://github.com/srsrb/VISION`

6 novembre 2024

Le but de ce TP est d'implémenter l'algorithme de Graph Cuts pour calculer une carte de disparité à partir de deux images stéréoscopiques. Cette carte permet d'estimer la profondeur des objets présents dans la scène, en exploitant les différences entre les images pour reconstituer une représentation 3D.

1 Usage

Après l'exécution, une fenêtre contenant les deux images apparaîtra :

FIGURE 1 – Les deux images d'entrée

Il faudra patienter plus ou moins longtemps en fonction des valeurs des paramètres définis pour construire le graph et calculer le flot maximum. Une fois la carte de disparité calculée, elle s'affichera sur l'image de gauche comme ceci :

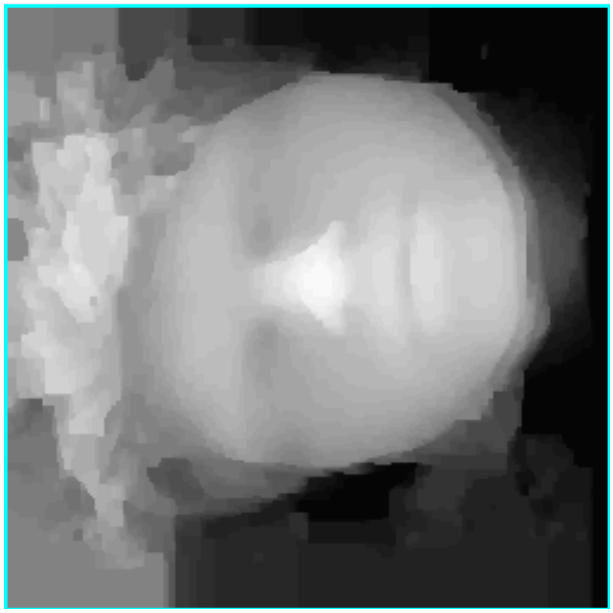

FIGURE 2 – La carte de disparité calculée

Ensuite, il suffira de faire un clique gauche pour flouter la carte de disparité :

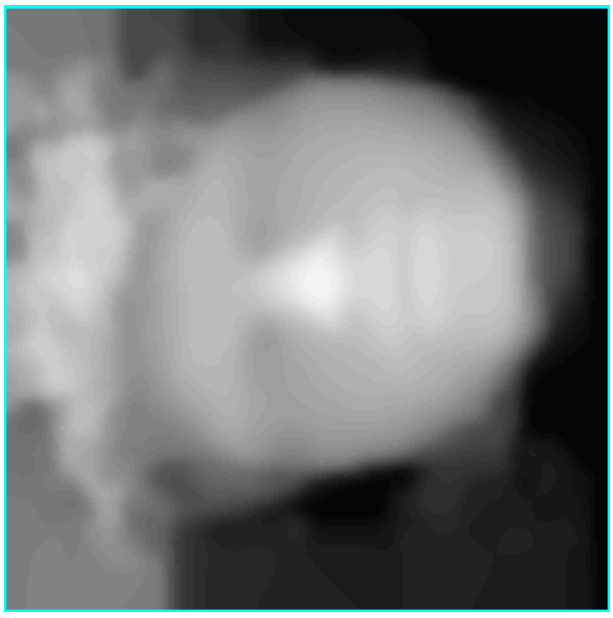

FIGURE 3 – La carte de disparité floutée

Enfin, un dernier clique gauche fera ouvrir une nouvelle page avec le rendu du maillage 3D :

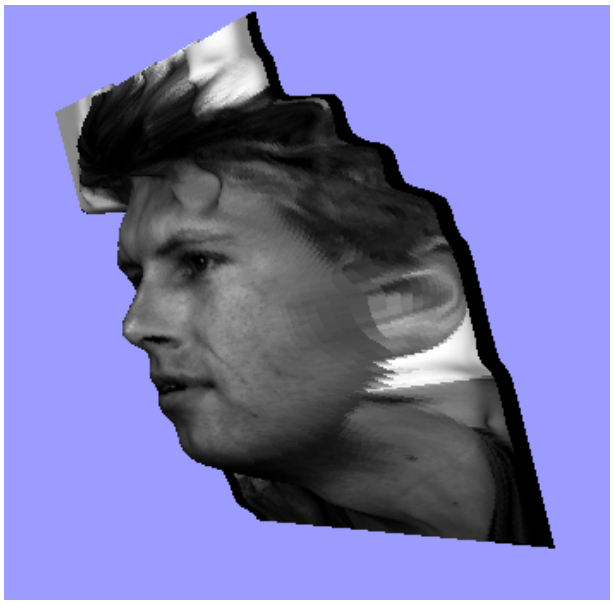

FIGURE 4 – 3D mesh rendering

Ensuite, il suffit de faire un clique droit et le programme se terminera.

2 Impact des paramètres sur les résultats

Dans cette section, nous analysons l'impact des différents paramètres sur la qualité des résultats.

2.1 Paramètre λ

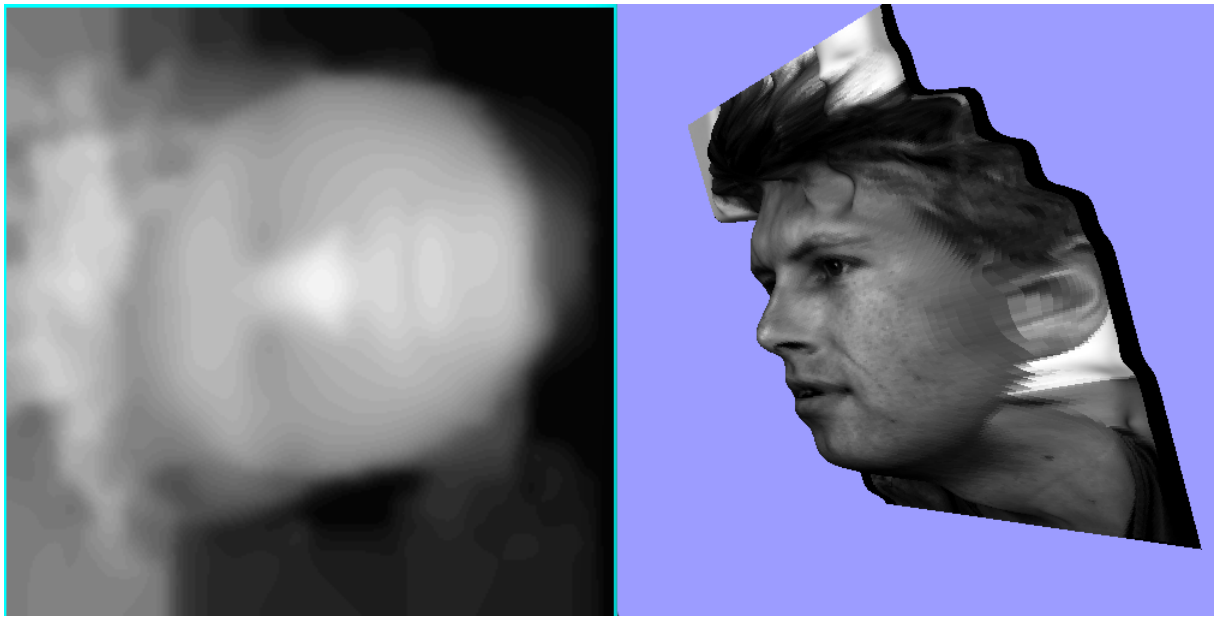

FIGURE 5 – Carte de disparité floutée et 3D mesh rendering avec lambda = 0.1

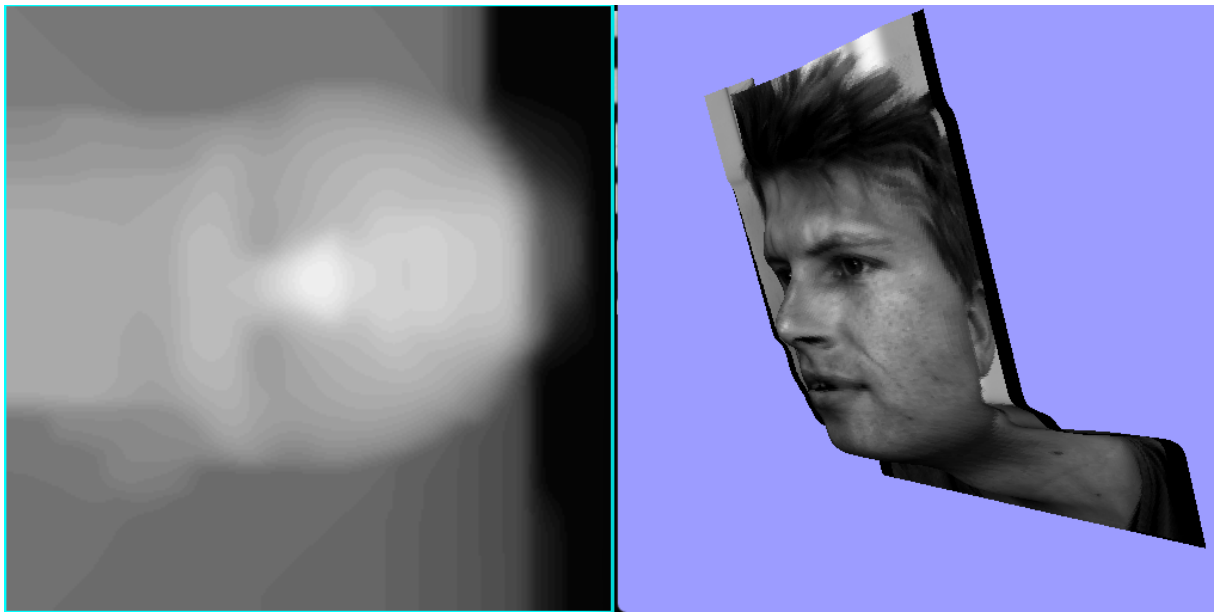

FIGURE 6 – Carte de disparité floutée et 3D mesh rendering avec lambda = 0.5

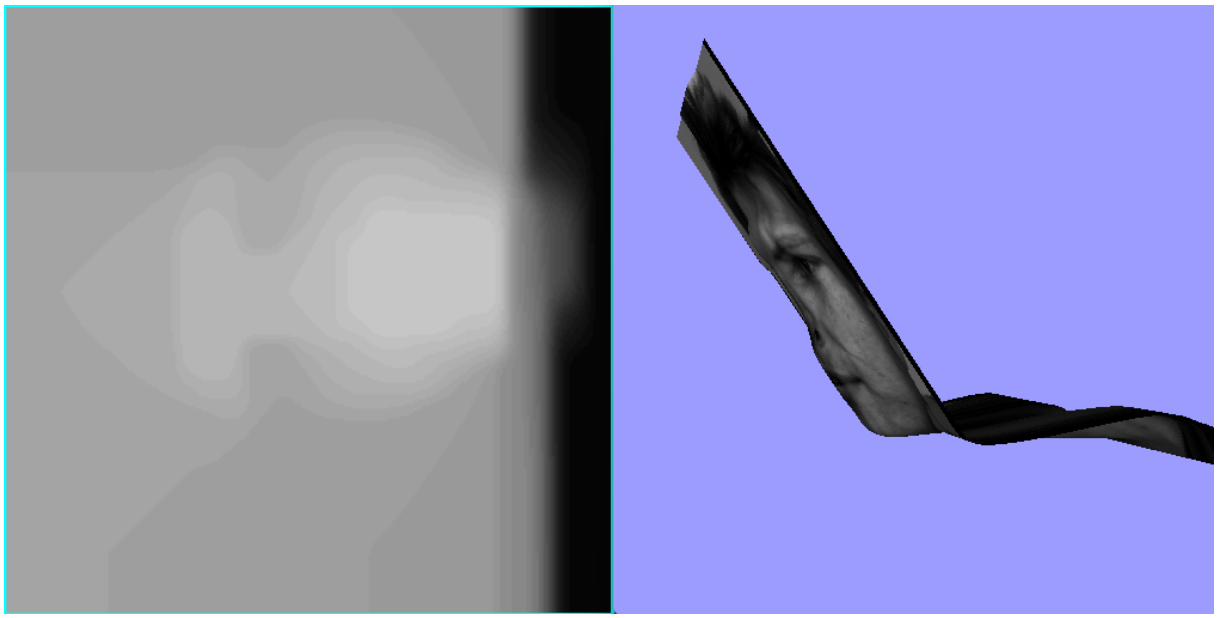

FIGURE 7 – Carte de disparité floutée et 3D mesh rendering avec lambda = 1

Ce paramètre contrôle la force de "lissage". le paramètre λ permet d'éviter des cartes de disparité trop bruyantes en lissant les disparités dans les zones homogènes. Plus on augmente λ, plus le lissage sera puissant, ce qui peut entraîner une perte de détails fins dans les zones de forte variation. À l'inverse, une valeur trop faible pourrait mener à une carte de disparité trop bruitée. Plus λ est grand, plus il augmente le temps d'execution.

2.2 Paramètre NCC neighborhood size (win)

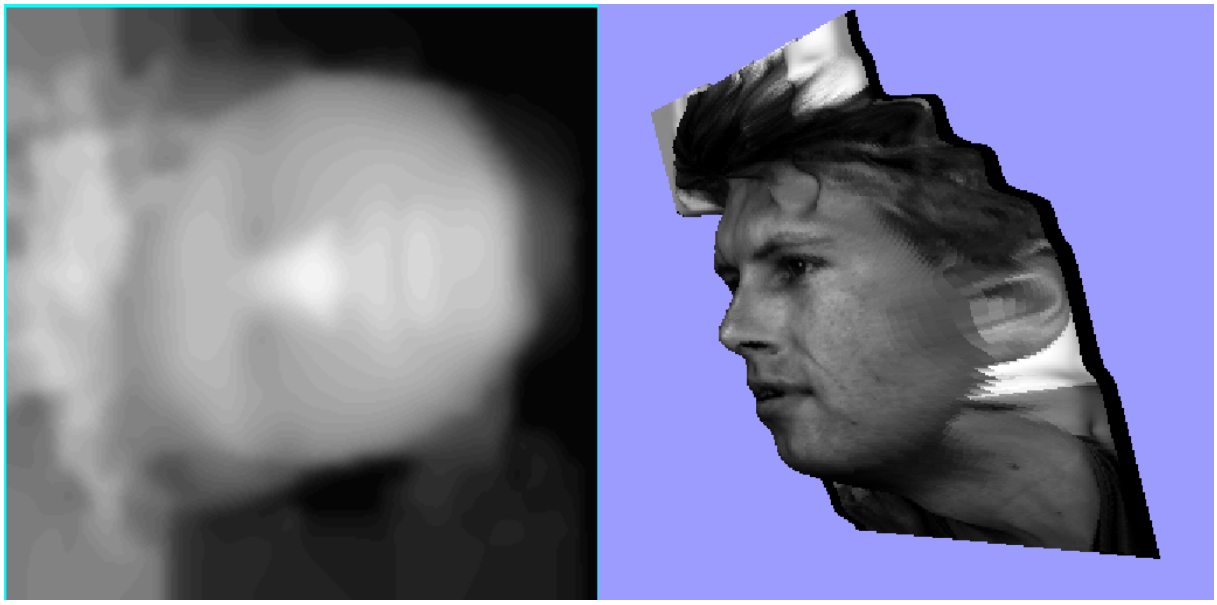

FIGURE 8 – Carte de disparité floutée et 3D mesh rendering avec win = 3

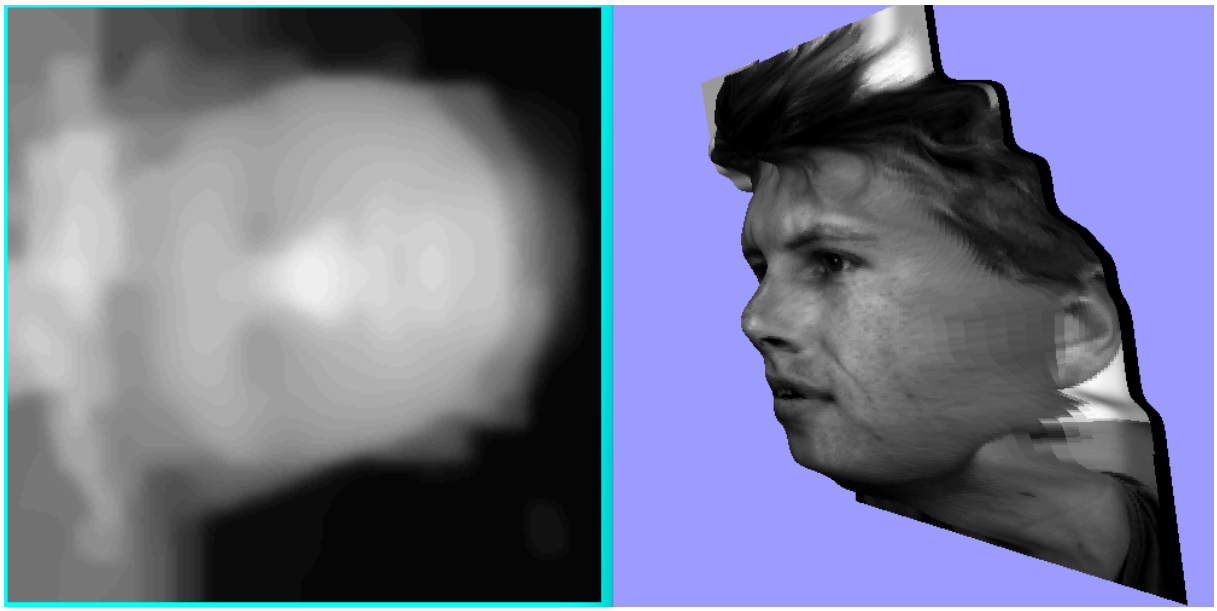

FIGURE 9 – Carte de disparité floutée et 3D mesh rendering avec win = 10

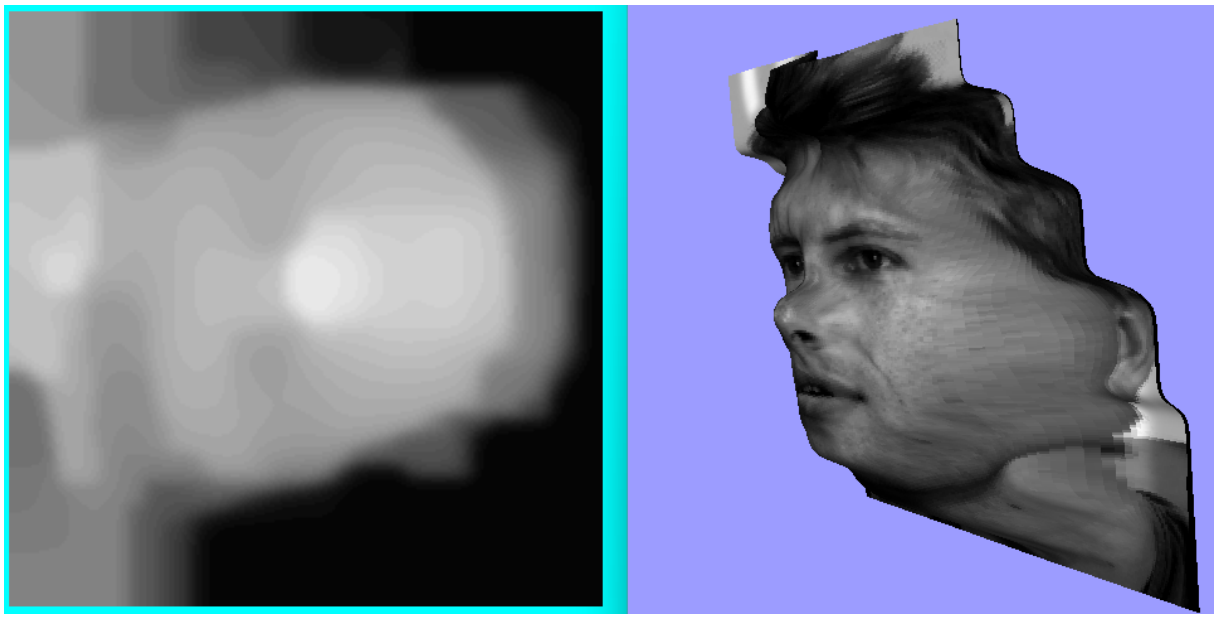

FIGURE 10 – Carte de disparité floutée et 3D mesh rendering avec win = 20

La taille de la fenêtre de corrélation est un facteur important. La variable win définit la taille de la fenêtre utilisée pour calculer la similarité entre les pixels correspondants dans les deux images. Plus la fenêtre est grande plus elle capture les détails locaux mais augmente la lenteur de l'algorithme. En augmentant la taille de la fenetre utilisée, on va perdre en détails, comme on peut l'observer dans la figure 10.

3 Comparaison avec l'algorithme "region-growing"

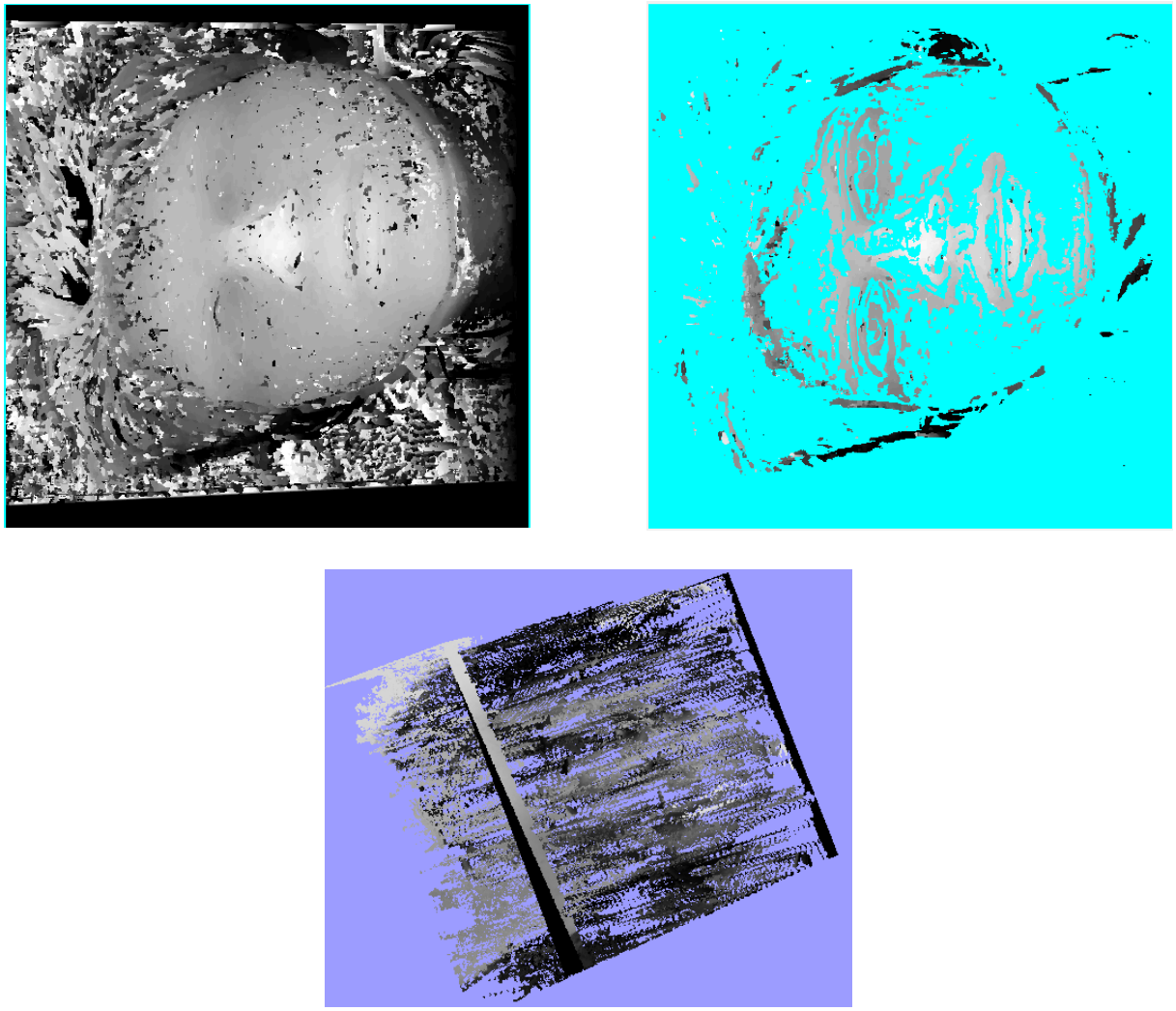

FIGURE 11 – Algortihme "region growing"

L'algorithme de Graph Cut est plus complexe et offre des résultats plus robustes et optimaux pour des problèmes de correspondance de disparité, tandis que la méthode de Region Growing est plus simple et rapide, mais moins précise pour des images avec des différences complexes. Je n'ai pas réussi à faire fonctionner la méthode de "region-growing" sur nos images. Il est certain qu'avec un meilleur choix de paramètre, on pourrait s'améliorer et obtenir un résultat plus convenable. Cependant, ce résultat restera plus interessant avec une méthode de Graph Cut.