Université du Québec
École de technologie supérieure
Département de génie logiciel et des T.I.

Sujets émergeant en technologie de l'information

Rapport de Laboratoire

Numéro du laboratoire	2
Nom du laboratoire	Capture, traitement et affichage d'images 3D
Étudiant(s)	Simon-Olivier Harel Patrick Lavallée
Code(s) permanent(s)	HARS10068806 LAVP12048408
Numéro d'équipe	
Cours	GTI780
Session	E2017
Groupe	1
Chargé(e) de laboratoire	Louiza Oudni
Date	25-06-2017

Table des matières

Introduction

Actuellement le marché de l'imagerie 3D se voit en expansion via de nouveaux intérêts provenant d'une demande de plus de réalisme du monde dans un écran tel que pour le cinéma, les jeux par ordinateur, la médecine ainsi que pour le domaine de la vision par ordinateur tel que comprendre les scènes automatiquement. L'intérêt des consommateurs est motivé par le progrès des technologies et des techniques afin de recréer des images en trois dimensions.

L'objectif de ce second laboratoire est d'implémenter l'algorithme du *Depth-Image-Based Rendering* (DIBR) pour synthétiser une nouvelle image afin d'obtenir une paire d'images stéréoscopiques et de les afficher adéquatement sur la télévision 3D Dimenco. Pour ce faire, l'utilisation du premier laboratoire a été utilisée et modifier afin de s'adapter aux exigences de résolution de l'écran de la télévision et du format du message d'en-tête de pour l'écran.

Les sections à venir traiteront de l'implémentation de l'algorithme du DIBR et de son fonctionnement ainsi que des paramètres utilisés afin de générer l'image synthétisée.

Explication de l'algorithme DIBR

Intérêt de l'utilisation du DIBR

Les techniques de représentation de l'information visuelle 3D des images sont classifiées en trois types de représentations. Une première catégorie est axée sur la géométrie, c'est-à-dire que l'information géométrique fait partie des données décrivant la scène, communément appelée la géométrie explicite.

La seconde classification est basée sur les images, c'est-à-dire que la représentation de la scène utilise des images, soit l'intensité lumineuse arrivant à la caméra, au lieu de primitives géométriques. Cette classification se nomme sans géométrie.

La dernière catégorie, soit la géométrie implicite est une méthode hybride des deux premières catégories. Elle ajoute une représentation de l'information de profondeur de façon directe, mais préserve l'information de texture. C'est dans cette catégorie que l'algorithme du DIBR se situe.

Fonctionnement de l'algorithme du DIBR

Cet algorithme est constitué de 6 étapes distinctes qui seront présentées ci-dessous. La figure 1 suivante illustre l'idée générale du comportement.

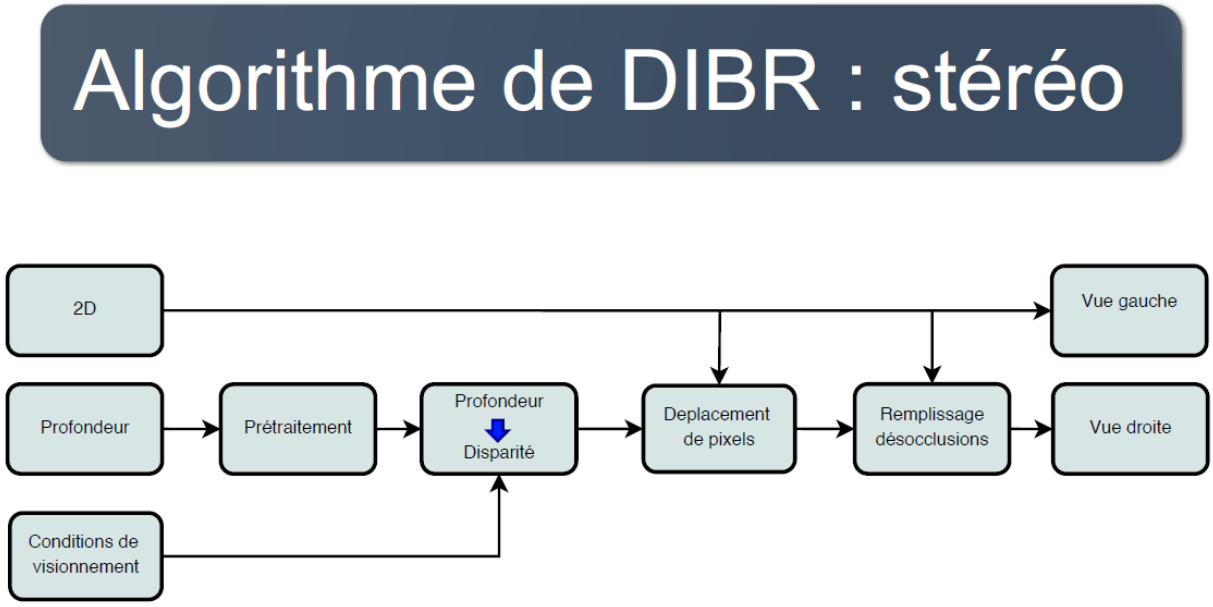


Figure 1 : Représentation de l'algorithme de DIBR.

Étape de prétraitement de la profondeur :

Cette étape consiste en la réduction du bruit dans l'image de profondeur et la détection des frontières. Ces étapes ont été effectuées lors du premier laboratoire.

Pour ce faire, l'utilisation du filtre smootGaussian a été mise de l'avant. L'utilisation d'un filtre passe-bas temporel se fait pour l'amélioration de la stabilité du mouvement des objets de la scène.

Remplissage de l'information manquante : Inpainting : utiliser l'information des bordes pour remplir l'intérieur de la région manquante

- Blocs d'exemple
- Inondation morphologique (floodfill)
- Variationnels

Mapping de la profondeur dans l'espace de l'observateur :

Cette étape consiste à calculer le positionnement des pixels en profondeur en fonction des dimensions de l'écran.

1. Détermination des dimensions de la télévision :

Sachant que la télévision à une distance diagonale **T** de 65.0 pouces et une résolution de **2160 : 3840** qui est équivalent à **9 : 32** et que par hypothèse nos posons que **Knear** soit de 10% et que **Kfar** soit de 20%.

Calculs :

$$T^2 = H^2 + W^2$$

Où

T : est la dimension diagonale de l'écran qui est 65.0 pouces qui est équivalant à 1651 mm
H : est la dimension en hauteur de l'écran en mm.
W : est la dimension en largeur de l'écran en mm.

$$\frac{H}{W} = \frac{9}{32}$$

$$H = \frac{9}{32} * W$$

$$T^2 = (\frac{9}{32} * W)^2 + W^2$$

$$T^2 = \frac{1105}{1024} * W^2$$

$$W = \sqrt{\frac{(1651^2 * 1024)}{1105}}$$

$$W = 1589.34\, mm$$

$$H = 447.00\, mm$$

2. *Calcul de la profondeur du point :*

Cette étape consiste à transformer la valeur d'un pixel de profondeur en format de l'espace de l'écran. Une transformation linéaire a été appliquée par la formule suivante.

$$Z_p = W\left(\frac{m}{2^N - 1} * \left(K_{near} + K_{far}\right) - K_{far}\right)$$

Où, Z_p : est la profondeur du point en millimètre.
m : est la valeur du pixel de profondeur en millimètre obtenu par la *Kinect*
N : est le nombre de bits représentant la profondeur soit 8.

L'image suivante explicite le *mapping* linéaire :

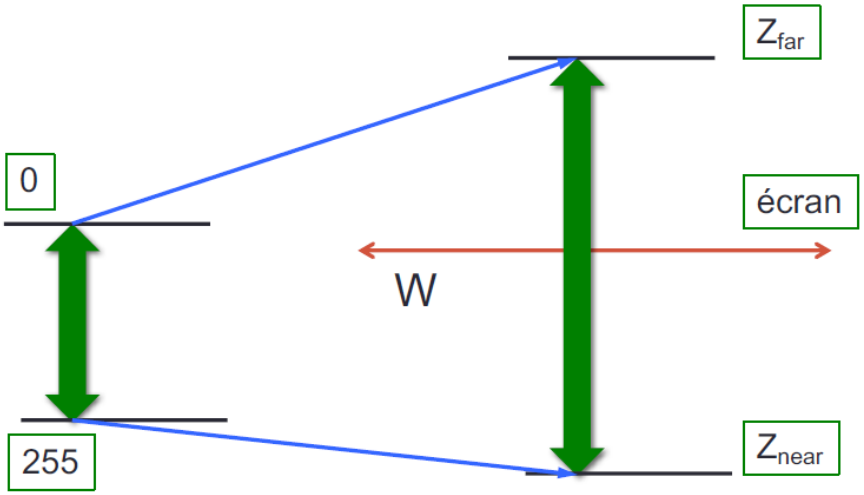


Figure 2: Mapping linéaire.

Étape de conversion de la profondeur en disparité

Cette étape consiste à calculer de disparité en mm

$$p = t_c * (1 - \frac{D}{D - z_p})$$

Où

t_c est la distance interoculaire tc = 65 mm
D est la distance entre l'observateur et l'écran D = 3*H
P valeur de la disparité à l'écran en mm.

Calcul de disparité en pixel

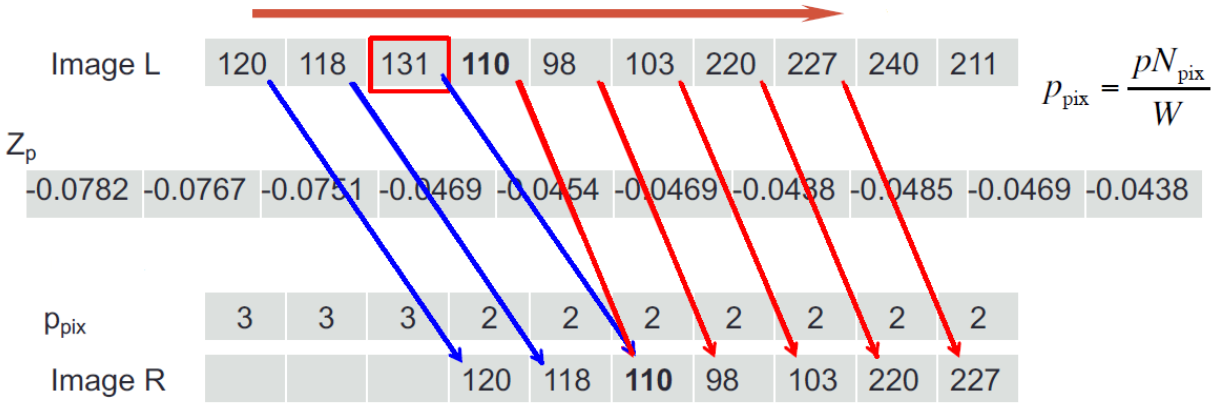
$$p_{pix} = p * (\frac{N_{pix}}{W})$$

Où

N_{pix} est le nombre de pixels sur la largeur de l'écran
P_{pix} valeur de la disparité à l'écran en pixel.

Déplacement de pixels vers la deuxième image

Calculs du déplacement du pixel :
D_{pix} = arrondir la valeur de ppix.



Remplissage des régions nouvellement exposées par les désocclusions

Un deuxième traitement *InPainting* peut être utilisé lors de ce processus. Par contre, le présent laboratoire ne l'applique pas.

Post traitement de l'image finale

Afin d'avoir une image de haute qualité, il est préférable d'ajouter un filtrage le long des frontières permettant un raffinement plus élevé de l'image de synthèse.

Résumé des paramètres utilisés

Tableau des différents paramètres et résultats

Paramètres	Signification	Valeurs
T	Dimension de diagonale de l'écran	1651.00 mm
H	Hauteur de l'écran	447.00 mm
W	Largeur de l'écran	1589.34 mm
	Résolution de l'écran	9 : 32
K_{near}	Distance devant l'écran	10%
K_{far}	Distance derrière l'écran	20%
Z_p	La profondeur du point en millimètre	Différent pour chaque pixel, valeur en mm.
N	Nombre de bits représentant la profondeur	8
m	Valeur du pixel de profondeur obtenue par la *Kinect*	Différent pour chaque pixel, valeur en mm.
t_c	Distance interoculaire	65 mm
D	Distance entre l'observateur et l'écran	1341 mm
P	Disparité à l'écran	Différent pour chaque pixel, valeur en mm.
P_{pix}	Disparité en pixel	Différent pour chaque pixel, valeur en pixel.
D_{pix}	Déplacement du pixel	Différent pour chaque pixel, valeur en mm.

Exemple de code

La section suivante met de l'avant les parties névralgiques du code qui ont permis de réaliser le laboratoire.

Génération de la nouvelle entête

Une méthode a été ajoutée à la fabrique d'images d'en-tête permettant d'instancier un objet de type *TopAndDownHeader*. Tout comme son prédécesseur, celle-ci détient la valeur du message H qui est a décodé

```
public static StereoscopicHeader Create(HeaderType headerType)
{
    switch(headerType)
    {
        case HeaderType.SideBySide:
            return CreateStereoscopicHeader() as SideBySideHeader;
        case HeaderType.TopAndDown:
            return CreateTopAndDownHeader() as TopAndDownHeader;
        default:
            throw new ArgumentException("HeaderFactory.Create : Unrecognized HeaderType requested");
    }
}
```

Figure 3 - Fabrique d'en-têtes

```
private static StereoscopicHeader CreateTopAndDownHeader()
{
    // The calculated Header Message for Top frame with color and
    // bottom frame processed throught the DIBR algorithm.
    var H = "F10140801000D47C48BCF22233000000000000000000000000000000AF7AB8ED";
    return new TopAndDownHeader(H);
}
```

Figure 4 - Instanciation de la nouvelle entête

Initialisation des paramètres de la télévision

Les variables nécessaires au calcul ont été ajoutées à la classe *MainWindow*

```
// DIBR algorithm parameters
// Measurement from the Dimenco TV
private double T = 1651; //Should be in mm.
private double W = 0.0;
private double H = 0.0;
private int TV_Ratio_Height = 9;
private int TV_Ratio_Width = 16;

private const int NUMBER_PIXELS_WIDTH = 3840

// User distance from Dimenco TV
private double D = 0.0;
private double tc = 65; // milimeters
private double knear = 0.1;
private double kfar = 0.2;
```

Figure 5 - Variables du problème

Une méthode privée à la classe permet d'initialiser les valeurs dimensionnelles de la télévision Dimenco.

```
private void SetTelevisionDimension()
{
    H = Math.Sqrt(Math.Pow(T, 2) * Math.Pow(TV_Ratio_Height, 2) /
        (Math.Pow(TV_Ratio_Width, 2) + Math.Pow(TV_Ratio_Height, 2)));

    W = Math.Sqrt(Math.Pow(T, 2) - Math.Pow(H, 2));

    D = 3 * H;
}
```

Figure 6 - Initialisation des dimensions de la télévision

Implémentation DIBR

Suite au traitement de la profondeur fait lors de la première itération, une nouvelle image de synthèse doit être créée en utilisant la profondeur et l'information de couleur. La grosseur du flux de couleur est déterminée sur-le-champ.

```
private byte[] GenerateNewImage(ColorFrame colorFrame, byte[] depthValues)
{
    // The total size of the color stream, multiplied by a factor of 4 because of the BGRA format
    var colorSize = RAWCOLORHEIGHT * RAWCOLORWIDTH * 4;
```

Figure 7 - Signature de méthode GenerateNewImage

L'information de couleur originale est ensuite copiée dans deux tableaux d'octet. Le premier servira a savoir l'index courant des pixels en traitement et le second contiendra l'information de l'image de synthèse une fois la disparité appliquée aux pixels sources.

```
// Source color image
var leftImage = new byte[colorSize];
colorFrame.CopyConvertedFrameDataToArray(leftImage, ColorImageFormat.Bgra);

// The constructed image to return, initialized with original color data (without disparity)
var newImage = new byte[colorSize];
colorFrame.CopyConvertedFrameDataToArray(newImage, ColorImageFormat.Bgra);
```

Figure 8 - Initialisation de l'image de synthèse

Finalement l'algorithme est appliqué pour chacun des pixels :

1. Chaque pixel de couleur doit être traité par incrément de 4 octets (BGRA = 1 octet chaque)
2. La valeur de Zp est déterminée utilisant la valeur de profondeur du pixel courant
3. La disparité entre la source et la destination est calculée
4. La même disparité doit être appliquée à chaque octet du format de l'image de destination

```
for (int colorIndex = 0; colorIndex < colorSize; colorIndex += 4)
{
    // Calculating Zp
    var zp = W * ((depthValues[colorIndex] / byte.MaxValue) * (knear + kfar) - kfar);

    // Calculating Disparity
    var p = tc * (1 - (D / (D - zp)));
    var disparity = Convert.ToInt32(Math.Round(p * 1920 / W));

    // Applying the same disparity to each subsequent 4 bytes representing the BGRA co
    for (int bgraIndex = 0; bgraIndex < 4; bgraIndex ++)
    {
        var pixelPosition = colorIndex + bgraIndex;
        var newPixelPosition = pixelPosition + disparity * 4;

        // Sanity check to ensure the new pixel position falls within the range of the
        if (newPixelPosition >= 0 && newPixelPosition < newImage.Length)
        {
            newImage[newPixelPosition] = leftImage[pixelPosition];
        }
    }
}

return newImage;
```

Figure 9 - Implémentation de DIBR

Conclusion

En somme, ce laboratoire a permis à l'équipe d'expérimenter l'algorithme DIBR. La conception faite lors de la première itération c'est montré souple et aucune modification majeure n'a dû être apportée lors de l'implémentation des nouvelles fonctionnalités.

De plus, les objectifs ont été atteints. Le nouveau message d'en-tête permet d'avoir facilement une image d'en-tête pour le nouveau format d'image. En utilisant l'image de profondeur générée lors du premier laboratoire, une image couleur de synthèse est générée suivant l'algorithme DIBR. La qualité de l'image résultante est assurée en partie par l'amélioration de l'information de profondeur initiale.

Les concepts mis de l'avant dans ce laboratoire permettent de comprendre les enjeux quant à l'industrie de l'imagerie stéréoscopique. Il s'agit d'un domaine encore en émergence qui à terme aura des impacts sur le monde télévisuel et du cinéma.