



**POLYTECHNIQUE  
MONTREAL**

UNIVERSITÉ  
D'INGÉNIERIE

**INF6804 – Vision par ordinateur**

**Hiver 2019**

**TP2 – Description et comparaison de régions d'intérêt**

**Groupe 01**

**1743175 – Bilal Itani**

**1740543 – Mehdi Kadi**

**Soumis à :**

**Hui-Lee Ooi**

**Soufiane Lamghari**

**1 mars 2019**

---

## Table des matières

<b>1. Présentation des deux approches à étudier</b>	<b>2</b>
1.1 Méthode de description par histogramme de gradients orientés (HOG)	2
1.2 Méthode de description basée sur les patrons locaux binaires (LBP)	3
<b>2. Hypothèses de performance dans des cas spécifiques</b>	<b>5</b>
<b>3. Description des expériences et bases de données</b>	<b>6</b>
<b>4. Description des deux implémentations utilisées</b>	<b>7</b>
4.1 Méthode par histogrammes de gradients orientés (HOG)	7
4.2 Méthode de description basée sur les patrons locaux binaires (LBP)	8
<b>5. Présentation des résultats de tests</b>	<b>9</b>
5.1 Résultats obtenus pour le cas de l'occlusion - Sequence Face	9
5.2 Résultats obtenus pour le cas de variation de l'ombrage - Sequence Sunshade	10
5.3 Résultats obtenus pour le cas d'une caméra et d'un sujet dynamique - Sequence Trellis	11
<b>6. Discussion des résultats et retour sur les hypothèses</b>	<b>13</b>
6.1 Cas de l'occlusion	13
6.2 Cas de variation de l'ombrage	14
6.3 Cas d'une caméra et d'un sujet dynamique	15
<b>7. Bibliographie</b>	<b>16</b>

## 1. Présentation des deux approches à étudier

Dans le cadre de ce travail, nous avons eu la chance d'explorer deux méthodes de description de régions d'intérêt dans une séquence vidéo afin de déterminer la meilleure dans certaines circonstances. La première méthode est basée sur la description par histogramme de gradients orientés. La deuxième méthode, appelée LBP, extrait des descripteurs de patrons locaux binaires. Afin d'évaluer ces méthodes, nous avons choisi de les utiliser dans un contexte de suivi. Afin de faciliter l'implémentation de notre expérience, nous avons choisi de faire du suivi de visage, puisqu'il existe déjà des implémentations dans OpenCV.

### 1.1 Méthode de description par histogramme de gradients orientés (HOG)

Selon l'article des auteurs de la méthode HOG [13], une *patch* est, d'abord, extraite d'une image et est redimensionnée à une taille 64 x 128. Ensuite, on calcule les gradients de la *patch*, sachant qu'il existe plusieurs noyaux pour extraire les gradients de la *patch* (i.e, le noyau de Sobel peut être un bon exemple). Une fois les gradients extraits, on peut calculer sa magnitude et son orientation à l'aide des formules ci-dessous :

$$g = \sqrt{g_x^2 + g_y^2}$$
$$\theta = \arctan \frac{g_y}{g_x}$$

Ici  $g_x$  représente le gradient en  $X$  obtenu à l'aide du noyau selon l'orientation de  $X$ ,  $g_y$  est le gradient selon  $Y$ ,  $g$  représente la magnitude et  $\theta$  est l'orientation du gradient. Une fois les gradients et orientations obtenus, la *patch* de départ est encore divisée en cellules de taille 8 x 8 et un histogramme d'orientation du gradient est calculé sur chacune des cellules 8 x 8. La figure 1, ci-dessous, montre l'approche utilisée pour le calcul de cet histogramme.

Ce qu'il faut retenir, c'est que les orientations considérées sont de 0 à 180 degrés. Un gradient avec un angle de 360 degrés est équivalent à un angle de 180 degrés. L'histogramme utilisé subdivise les angles avec un intervalle de 20 degrés. Ainsi, l'histogramme est obtenu à l'aide d'une pondération de l'orientation et de la valeur de la magnitude. Par exemple, sur la figure 1, ci-bas, il est possible de constater que la valeur 4 encadrée en rouge vaut 2 pour l'intervalle 0 et 2 pour l'intervalle 20. En effet, tel qu'indiqué par la direction de ce gradient (10), il est à mi-chemin entre 0 et 20. Une fois l'histogramme calculé, une normalisation est appliquée sur celui-ci afin qu'il soit insensible aux variations de lumière et de contraste. On concatène finalement tous les histogrammes calculés en un vecteur et c'est ce vecteur qui décrira l'ensemble de la *patch* de l'image. On fait ceci pour chaque *patch* de l'image, à différentes échelles et positions. Une comparaison est ensuite appliquée entre les vecteurs de caractéristiques (*features*, en anglais) pour identifier un objet.

Dans le cadre de ce laboratoire, nous allons utiliser la librairie DLIB qui contient notamment un détecteur de visage basé sur un histogramme d'orientation du gradient. Nous évaluerons cette méthode dans un contexte de suivi.

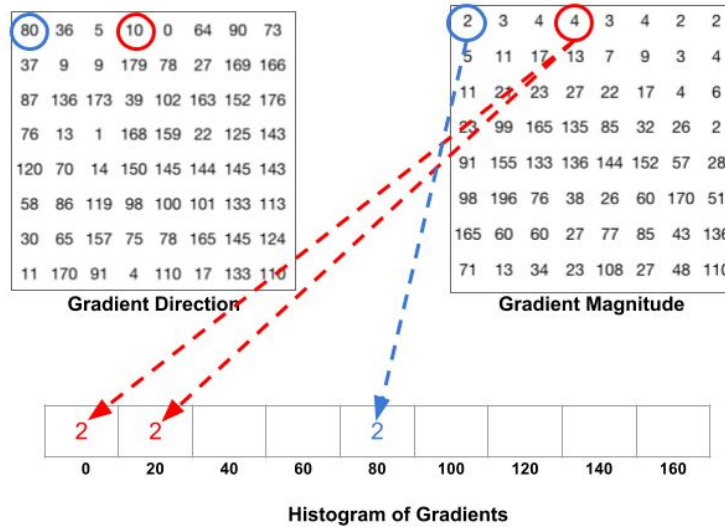


Fig. 1 : Calcul de l'histogramme de gradient

## 1.2 Méthode de description basée sur les patrons locaux binaires (LBP)

La méthode de description à l'aide de patrons locaux binaires permet de décrire une texture locale autour d'un pixel à l'aide de tests binaires. Il s'agit d'un histogramme de codes binaires sur un voisinage local de taille  $n \times n$ . On effectue, d'abord, une transformation de l'image en ton de gris, si elle ne l'est pas déjà, puis un seuillage d'un nombre  $P$  pixels du voisinage d'un pixel par la valeur de ce dernier [13]. Le résultat de ce seuillage est un nombre binaire ayant  $2^P$  valeurs possibles. Les formules suivantes indiquent comment est calculé le nombre binaire représentant LBP :

$$LBP_{P,R}(x_c, y_c) = \sum_{p=0}^{P-1} s(g_p - g_c) 2^p$$

$$\text{avec } s(x) = \begin{cases} 1 & x \geq 0 \\ 0 & x < 0 \end{cases}$$

Il convient de noter que  $g_c$  est la valeur d'intensité du pixel du centre  $(x_c, y_c)$  et  $g_p$  est la valeur d'intensité de  $P$  pixels espacés également sur un cercle de rayon  $R$ . La figure 2, ci-dessous, montre le processus de génération du descripteur LBP sur un cercle de rayon  $R$ .

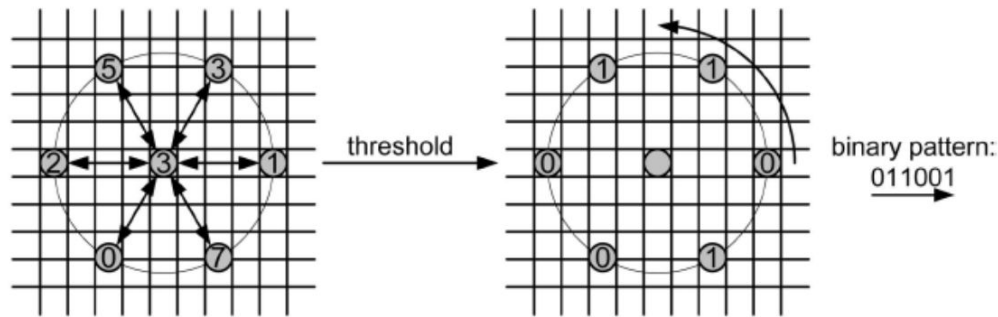


Fig. 2 : Génération du descripteur LBP

Afin d'être robuste au bruit, une variante de LBP introduit une constante  $a$ . Ainsi, le calcul de la fonction  $s$  devient comme suit :

$$s(g_p - g_c + a)$$

Une fois les codes binaires calculés, on calcule l'histogramme de ces derniers qui constitue notre descripteur tel que montré dans la figure 3 ci-dessous :

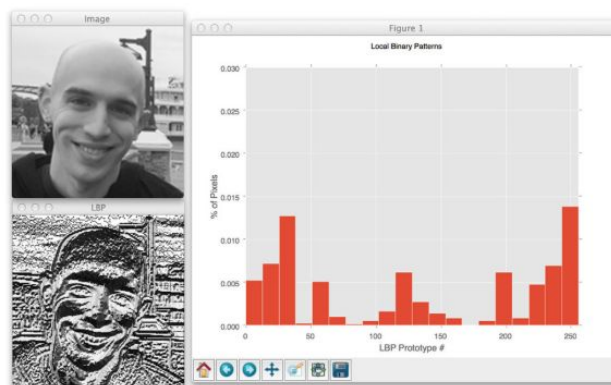


Fig. 3 : Calcul de l'histogramme LBP

LBP permet de capturer des détails très fins sur l'image, à un point que ceci peut être vu comme un défaut de LBP. Ainsi, une variation de l'échelle peut grandement affecter LBP. Il est donc important de choisir adéquatement le rayon  $R$  du cercle pour calculer LBP. Un petit rayon permet de relever des détails importants dans l'image alors qu'un rayon plus grand les éliminerait. Par conséquent, LBP n'est pas robuste, soit invariant à la mise à l'échelle et à la rotation. Il permet, toutefois, de bien identifier un objet avec précision si cet objet est capturé avec un même point de vue. LBP est souvent utilisé pour de la reconnaissance de visage et il est très performant, puisqu'il n'est pas complexe à calculer.

Dans le cadre de ce laboratoire, nous utiliserons la librairie *OpenCV* pour évaluer LBP. En effet, cette librairie contient déjà une implémentation de ce descripteur dans le but de faire de la reconnaissance faciale. Il sera facile, pour nous, de comparer LBP à HOG dans le cas d'utilisation du suivi d'un individu.

## 2. Hypothèses de performance dans des cas spécifiques

Dans le cas de l'occlusion, nous pensons que LBP sera plus performant que HOG puisque c'est un descripteur qui va se baser sur les détails de l'image. LBP est un descripteur local. Par conséquent, même si une partie du visage d'une personne est cachée par un objet, l'autre partie pourra être facilement reconnue à l'aide du descripteur; évidemment tout est une question du choix du voisinage. HOG est plutôt un descripteur ayant une vue plus globale de ce qu'il décrit. Ainsi, lorsqu'un objet cache une partie du visage d'une personne, il ne sera pas en mesure de la détecter vu qu'il ne prend pas en considération les détails fins de l'image comme LBP.

Dans le cas de la variation des ombres, nous pensons que HOG sera plus performant que LBP, puisque ce dernier est très sensible contrairement à HOG. Les deux descripteurs sont normalisés afin d'avoir une certaine invariance au changement de luminosité et de contraste, mais le fait que LBP est plus sensible que HOG lui rendra la tâche plus difficile. De plus, le sujet dans la séquence vidéo se déplace dans la scène, ce qui rendra la tâche difficile à LBP qui est très sensible au mouvement et n'offre pas d'invariance à la rotation et à la translation.

Finalement, dans le cas d'une caméra qui bouge et d'un point de vue dynamique, nous avons la ferme conviction que HOG sera meilleur que LBP. En effet, LBP est très sensible puisque c'est un descripteur local contrairement à HOG qui est un descripteur à plus grande échelle. LBP et HOG ne sont pas invariants à la rotation, à la translation ou à la mise à l'échelle. Toutefois, vu que HOG est un descripteur à plus grande échelle, il performera mieux que LBP.

### 3. Description des expériences et bases de données

Pour nos expériences, nous avons choisi d'utiliser la base de données VOT. Toutes les expériences que nous avons effectuées proviennent des données de cette base. En effet, nous avons extrait trois cas d'utilisation qui nous semblaient pertinent dans le domaine du suivi, un cas d'occlusion, un autre qui implique la variation des ombres et un troisième avec la caméra qui se déplace.

La première expérience consiste à suivre le visage d'une femme qui cache plusieurs parties de son visage avec un livre. La femme est assise sur un fauteuil, elle est statique et la caméra ne bouge pas. Le contraste et la luminosité de la vidéo sont adéquats pour un cas normal. La séquence vidéo contient 415 trames et a une taille de 352 x 288 pixels, le nom de la séquence utilisée est "*Sequence face*" [2, 5].

La deuxième expérience consiste à suivre le visage d'un homme qui se déplace d'une zone ensoleillée à une zone ombrée et vice versa. La caméra est statique, mais l'homme qui se déplace d'une zone à l'autre est mobile. La séquence vidéo contient 172 trames et a une taille de 352 x 288 pixels. La séquence vidéo utilisée se nomme "*Sequence sunshade*" [3, 5].

La troisième expérience consiste à suivre le visage d'un homme qui se déplace d'un restaurant vers l'extérieur de ce dernier. La caméra suit l'individu, elle est donc mobile. L'homme passe par des zones ensoleillées et sombres. La séquence vidéo contient 569 trames de taille 320 x 240 pixels. Le nom de la séquence vidéo se nomme "*Sequence trellis*" [4, 5].

Les séquences vidéos utilisées sont toutes annotées. En effet, elles sont toutes accompagnées d'un fichier texte contenant la position d'un point du rectangle englobant ainsi que sa largeur et sa hauteur. Par conséquent, il sera très facile pour nous de prélever des statistiques quant à la performance de LBP ou de HOG dans les diverses situations.

## 4. Description des deux implémentations utilisées

### 4.1 Méthode par histogrammes de gradients orientés (HOG)

Pour la méthode HOG, nous utilisons la librairie *dlib* qui contient déjà une implémentation de la détection des visages humains à l'aide des histogrammes de gradients orientés [6,7,8]. En effet, la méthode *get\_frontal\_face\_detector* retourne un détecteur d'objet qui est configuré pour détecter les visages humains. Cette technique est basée sur l'article "*Histograms of Oriented Gradients for Human Detection*" [9]. Après l'obtention de cet objet, nous lisons le fichier contenant la vérité absolue des boîtes englobantes identifiant les visages des personnes à chaque trame de la vidéo. Ceci est nécessaire afin de pouvoir prélever les statistiques de performances.

Ensuite, pour chacune des trames de la vidéo, nous appliquons un redimensionnement de l'image pour lui attribuer une hauteur de 1 000 pixels et une largeur proportionnelle. Ceci nous permet de faire ressortir le plus de détails possible sur l'image pour pouvoir bien détecter les visages qui sont petits. Plus le redimensionnement est grand, plus la performance de l'algorithme chute, car il y a beaucoup plus d'histogrammes à calculer.

Finalement, on appelle la méthode "detector" qui va détecter les visages à l'aide des HOG. Si les boîtes englobantes des visages détectés font une intersection avec les boîtes englobantes de la vérité absolue, alors on procède au calcul de l'IoU. Parfois, aucune intersection n'est détectée, on considère ainsi un IoU de 0. Une fois les trames étudiées, nous effectuons un calcul de l'IoU minimum et maximum, celui de la médiane des IoU sur l'ensemble des trames, de la médiane des IoU sur l'ensemble des trames sur lesquelles un visage a été détecté. On effectue les mêmes calculs avec la moyenne. Finalement, on calcule aussi le nombre de trames sur lesquelles la méthode a détecté des visages.



## 4.2 Méthode de description basée sur les patrons locaux binaires (LBP)

Pour la méthode de description basée sur les patrons locaux binaires, nous utilisons un fichier classificateur LBP préentraîné. Ce classificateur vient sous la forme d'un fichier .xml qui a été généré lors de son entraînement à l'aide de la librairie *OpenCV*. En effet, ce classificateur est entraîné sur une centaine d'images contenant des visages humains, ceux-ci sont des exemples positifs qui sont tous réduits à une même taille, soit 20 x 20 pixels [10]. On entraîne aussi le classificateur avec des exemples négatifs. Ce classificateur va retourner "1" lorsque la région qu'il observe contient bien un visage et "0" autrement. *OpenCV* offre la possibilité de lire un classificateur à partir d'un fichier .xml à l'aide de la méthode `cv2.CascadeClassifier`. C'est grâce à ce classificateur [11] que nous pourrions reconnaître des visages avec LBP.

Tout comme dans l'implémentation de HOG, nous lisons un fichier texte contenant les boîtes englobantes de vérité absolue qui indique où sont les visages à chaque trame, car ceci sera nécessaire pour le prélèvement des statistiques. Chaque trame d'une séquence vidéo est convertie en ton de gris. On choisit le ton gris vu que LBP ne fonctionne que sur un canal. À l'aide du classificateur LBP, on appelle la méthode "detectMultiScale" qui permet d'utiliser ce classificateur à plusieurs échelles pour détecter un visage dans une région. Cette méthode retourne une liste de tuples contenant les positions  $x$  et  $y$  des points supérieurs gauches des rectangles englobants autour des visages détectés ainsi que leurs largeurs et hauteurs correspondantes.

Finalement, à l'aide de cette liste de tuples, nous effectuons le calcul de l'intersection entre les rectangles et ceux de la vérité absolue. Tout comme avec la méthode HOG, dans le cas où il y a intersection, nous calculons l'IoU, sinon, nous attribuons la valeur 0 à l'IoU. On effectue, aussi, le calcul de l'IoU maximal et minimal, un calcul de la médiane des IoU sur l'ensemble des trames, un calcul de la médiane des IoU sur l'ensemble des trames sur lesquelles un visage a été détecté. Nous reproduisons les mêmes calculs pour déterminer la moyenne. Finalement, on calcule aussi le nombre de trames sur lesquelles la méthode a détecté des visages.

## 5. Présentation des résultats de tests

Dans la présente section, nous allons présenter les résultats obtenus pour trois séquences vidéo que nous avons identifiées comme étant pertinentes. Les métriques que nous avons relevé sont :

- Moyenne et médiane des IoU sur toutes les trames (incluant ceux où il n'y a eu aucune détection)
- Moyenne et médiane des IoU sur les trames de détections (uniquement ceux où une détection à eu lieu)
- IoU minimal et maximal
- Taux de détection

En tout temps dans les séquences vidéos, les rectangles bleus sont les coordonnées de détection de la vérité absolue. Les rectangles verts sont ceux obtenus expérimentalement.

### 5.1 Résultats obtenus pour le cas de l'occlusion - Sequence Face

Pour le cas de l'occlusion, le tableau 1, ci-dessous, montre les divers résultats que nous avons obtenus :

Méthode	Moyenne IoU toutes les trames (%)	Médiane IoU toutes les trames (%)	Moyenne IoU trames de détection (%)	Médiane IoU trames de détection (%)	IoU minimal (%)	IoU maximal (%)	Taux de détection (%)
HOG	39.2	60.5	78.2	79.6	60.5	85.9	50.1
LBP	<b>56.1</b>	<b>63.9</b>	<b>64.9</b>	<b>64.3</b>	<b>51.4</b>	<b>74.7</b>	<b>86.5</b>

Tableau 1 : Différentes statistiques de performances sur la séquence vidéo “Sequence Face”

Les figures 4 et 5, ci-dessous, montrent respectivement la détection d'un visage à l'aide de la méthode HOG et LBP.

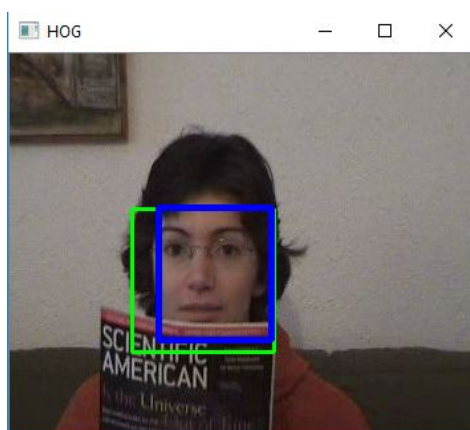


Fig. 4 : Détection d'un visage à l'aide de la méthode HOG sur une trame de “Sequence Face”

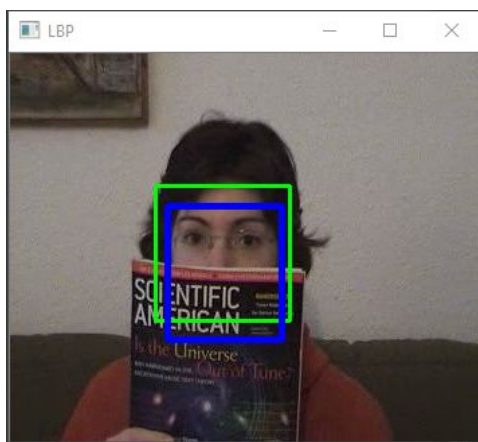


Fig. 5 : Détection d'un visage à l'aide de la méthode LBP sur une trame de “Sequence Face”

## 5.2 Résultats obtenus pour le cas de variation de l'ombrage - Sequence Sunshade

Pour le cas de variation de l'ombrage, le tableau 2, ci-dessous, montre les divers résultats que nous avons obtenus.

Méthode	Moyenne IoU toutes les trames (%)	Médiane IoU toutes les trames (%)	Moyenne IoU trames de détection (%)	Médiane IoU trames de détection (%)	IoU minimal (%)	IoU maximal (%)	Taux de détection (%)
HOG	46.0	47.2	47.7	47.4	28.1	64.9	96.5
LBP	26.5	0	61.6	62.4	44.7	73.5	43.0

Tableau 2 : Différentes statistiques de performances sur la séquence vidéo “Sequence Sunshade”

Les figures 6 et 7, ci-dessous, montrent respectivement la détection d'un visage à l'aide de la méthode HOG et LBP:

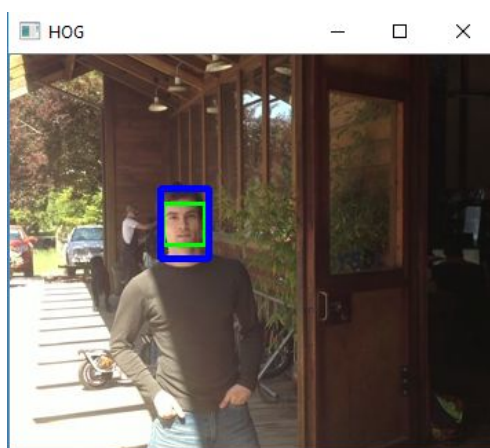


Fig. 6 : Détection d'un visage à l'aide de la méthode HOG sur une trame de “Sequence Sunshade”

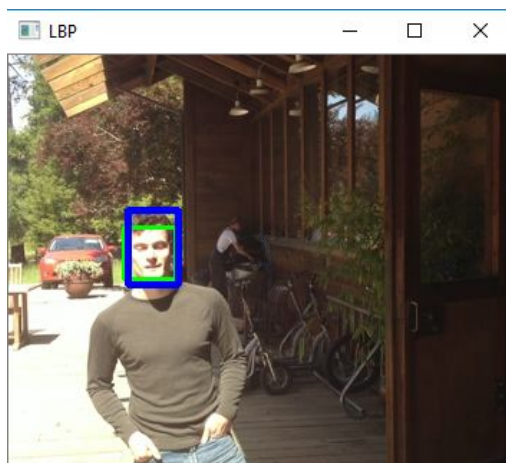


Fig. 7 : Détection d'un visage à l'aide de la méthode LBP sur une trame de “*Sequence Sunshade*”

### 5.3 Résultats obtenus pour le cas d'une caméra et d'un sujet dynamique - Sequence Trellis

Pour le cas d'une caméra et d'un sujet dynamique, le tableau 3 ci-dessous montre les divers résultats que nous avons obtenus :

Méthode	Moyenne IoU toutes les trames (%)	Médiane IoU toutes les trames (%)	Moyenne IoU trames de détection (%)	Médiane IoU trames de détection (%)	IoU minimal (%)	IoU maximal (%)	Taux de détection (%)
HOG	34	0	81.1	82.2	41.3	97.7	42.5
LBP	26.5	0	68.1	68.2	46.9	82.5	38.8

Tableau 3 : Différentes statistiques de performances sur la séquence vidéo “*Sequence Trellis*”

Les figures 8 et 9 ci-dessous montrent respectivement la détection d'un visage à l'aide de la méthode HOG et LBP :

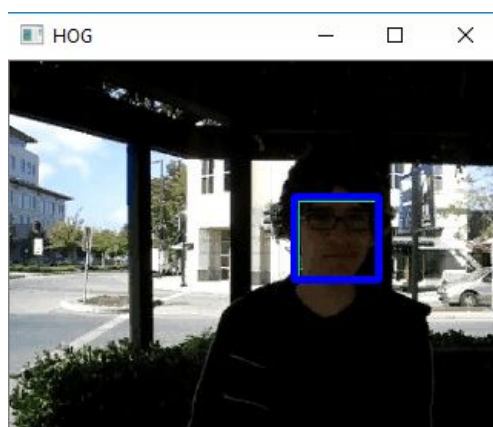


Fig. 8 : Détection d'un visage à l'aide de la méthode HOG sur une trame de “*Sequence Trellis*”

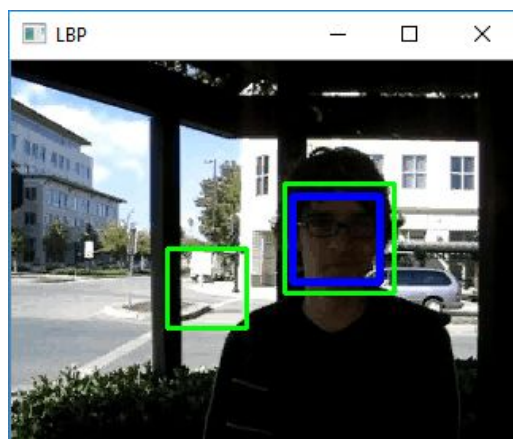


Fig. 9 : Détection d'un visage à l'aide de la méthode LBP sur une trame de "*Sequence Trellis*"

## 6. Discussion des résultats et retour sur les hypothèses

Dans cette section, nous discuterons des résultats obtenus et présentés à la section 5 du présent document et nous ferons un bref retour sur les hypothèses que nous avons émises.

### 6.1 Cas de l'occlusion

Pour le cas de l'occlusion, le tableau 1 montre les différentes statistiques que nous avons prélevées pour les deux méthodes. Pour HOG, il est possible de constater que le taux de détection est de 50.1%. Ainsi, HOG n'arrive pas à détecter le visage de la personne une trame sur deux, alors que ce dernier est toujours visible sur la séquence vidéo. Lorsque HOG détecte un visage, il le détecte en moyenne à 78.2% de IoU avec la vérité absolue, ce qui est très bien. La moitié des détections ont un IoU supérieur à 79.6%, ce qui est tout de même précis. Le IoU minimal de HOG est de 60.5% alors que le IoU maximal est de 85.9%. Ainsi, lorsque HOG détecte un visage, il le détecte très bien. Le souci avec HOG est qu'il ne regarde pas l'ensemble de l'image avec détail lorsqu'il construit son histogramme. Il a donc une vue un peu plus générale d'une *patch* de la vidéo, ce qui est problématique lorsque le sujet de la séquence vidéo cache son visage avec un article. HOG n'arrive tout simplement plus à détecter le visage bien que plus de la moitié de ce dernier est encore visible.

Pour le cas de LBP, il est possible de constater que le taux de détection est de 86.5% dans cette expérience, ce qui est largement supérieur à la méthode HOG. Lorsque LBP détecte un visage, il le détecte avec un IoU moyen de 64.9%. La moitié des détections ont un IoU supérieur à 64.3%, ce qui est bien, mais ce n'est pas meilleur qu'HOG, en général, HOG est plus précis lorsqu'il détecte un visage. Le IoU minimal de la méthode LBP est de 51.4% alors que le IoU maximal est de 74.7%, ce qui est inférieur à la performance d'HOG. Le taux de détection de LBP est largement supérieur à celui d'HOG. Ceci peut être expliqué par le fait que LBP est un descripteur qui s'attarde plus sur les détails de l'image contrairement à HOG qui est un peu plus général. Ainsi, lorsque le sujet cache son visage avec une revue, il est toujours possible pour LBP de voir les détails du reste du visage du sujet et de le détecter. Ceci n'est pas le cas avec HOG.

Bien qu'HOG a démontré de meilleurs résultats en terme de IoU, LBP est largement supérieur à HOG dans ce cas d'utilisation dans la mesure où LBP a eu un taux de détection de 86.5% contrairement à HOG qui a obtenu que 50.1%. HOG est plus précis lorsqu'il détecte, mais LBP est beaucoup plus adapté. Cette expérience confirme l'hypothèse que nous avons émises à la section 2 du présent document dans la mesure où LBP a eu un meilleur taux de détection, ce qui le rend plus performant que HOG. HOG est toutefois plus précis lorsqu'il détecte.

## 6.2 Cas de variation de l'ombrage

Dans le cas de la variation de l'ombrage, le tableau 2 montre les différentes statistiques que nous avons prélevées pour HOG et LBP. Pour ce qui est de HOG, son taux de détection est de 96.5%, ce qui est impressionnant. En effet, il a détecté le visage de l'individu pratiquement à chaque trame. Lorsqu'il y a eu détection, le IoU moyen est de 47.7%, ce qui n'est très précis. La médiane des IoU est de 47.4%, ainsi, la moitié des IoU sont supérieurs à 47.4%. L'IoU minimal est de 28.1% alors que l'IoU maximal est de 64.9%. On s'attendait à un meilleur résultat de la méthode HOG puisqu'elle a une vue globale et qu'elle est normalisée pour ne pas être affectée par la variation de luminosité ou de contraste.

Pour le cas de LBP, son taux de détection est largement plus bas que HOG. En effet, le taux de détection de LBP est de 43.0% contrairement à 96.5% pour HOG. Ceci est un comportement attendu puisque LBP est très sensible. Le fait que le sujet se déplace dans la séquence vidéo et qu'il y a cette variation d'ombrage n'aide pas du tout à la détection. En effet, LBP est un descripteur local qui est très sensible au détail dans une image. Il devrait avoir une certaine résistance aux variations de luminosité et de contraste puisqu'il est normalisé, mais le sujet dans la vidéo se déplace rapidement. Lorsque LBP détecte le visage du sujet, l'IoU est en moyenne de 61.6%, ce qui est supérieur à HOG, il est donc plus précis. La médiane des IoU est de 62.4%, ainsi, la moitié des IoU sont supérieurs à cette valeur, ce qui est très bien. L'IoU minimum de LBP est de 44.7% alors que le maximum était de 73.5%, ce qui est très bien.

Bien que LBP est largement plus précis que HOG au moment des détections, LBP ne détecte pas du tout de visage dans environ 60% des trames de la séquence vidéo contrairement à HOG qui a réussi à faire des détections dans presque 97% des trames. La différence ici est très grande puisque HOG a une vue plus globale de la *patch* de l'image qu'il traite contrairement à LBP qui dépend d'un rayon de voisinage. De plus, lorsque le sujet était dans la partie contenant de l'ombre dans la scène, LBP ne le détectait tout simplement pas. LBP ne le détectait que lorsque le sujet était dans la partie ensoleillée. HOG pouvait, par contre, le détecter autant dans la zone ensoleillée que dans la zone avec de l'ombre. Nos observations confirment les hypothèses que nous avons émises dans la section 2 du présent document à savoir qu'HOG sera plus performant que LBP puisque ce dernier est très sensible.

### 6.3 Cas d'une caméra et d'un sujet dynamique

Finalement, dans le cas d'une caméra et d'un sujet dynamique, le tableau 3 montre les différentes statistiques que nous avons prélevées pour HOG et LBP dans cette expérience. Pour HOG, il est possible de constater que le taux de détection du visage du sujet de la séquence vidéo a été détecté dans 42.5% des trames. Cela veut dire que dans presque 60% des cas, il n'a pas été possible de détecter son visage. Le IoU moyen lorsque le visage a été détecté est de 81.1%, ce qui est très bien. La médiane des IoU pour les trames où il y a eu détection est de 82.2%, donc la moitié des IoU est supérieure à cette valeur. L'IoU minimum est de 41.3% alors que l'IoU maximum est de 97.7%. Ainsi, il est possible de constater qu'HOG est très précis lorsqu'il réussit à détecter le visage du sujet. Par contre, son taux de détection est décevant, ceci est dû au fait que la caméra et le sujet se déplacent dans la séquence vidéo. Parfois, le sujet passe dans des zones moins ensoleillées alors que dans d'autres cas, il est dans des zones très illuminées. HOG n'est pas invariant à la rotation et comme la caméra se déplace aussi avec le sujet, les points de vue changent et il est plus difficile pour ce descripteur de détecter l'individu.

Pour la méthode LBP, son taux de détection est de 38.8%, presque aussi bas que HOG. Tout comme HOG, il n'est pas possible de détecter le sujet dans presque 60% des trames alors que son visage est toujours présent dans la séquence vidéo. Lorsque LBP détecte le visage du sujet, la moyenne des IoU est de 68.1%, la médiane est quant à elle de 68.2%. Ainsi, plus de la moitié des IoU est supérieure à cette valeur. Le IoU minimum était de 46.9% alors que le maximum était de 82.5%. Ce ne sont donc pas de mauvaises statistiques de précision pour LBP, mais le taux de détection est très bas, ce qui est décevant, mais explicable dans la mesure où que LBP est très sensible au détail et n'offre aucune invariance aux transformations de base d'une image. De plus, la figure 9 montre un fait qui arrive souvent dans cette séquence avec LBP, soit la détection de faux positifs, chose qui n'arrive pas avec HOG.

En sommes, HOG a été beaucoup plus précis que LBP lors de la détection du visage du sujet. Ceci est explicable dans la mesure où il a une vue un peu plus globale de la région où l'histogramme est construit que LBP. Ce dernier dépend d'un rayon de voisinage. Plus ce dernier est petit, plus il est sensible aux variations. Ceci confirme notre hypothèse à savoir qu'HOG sera plus performant que LBP. Bien que les deux méthodes n'ont pas bien performé sur cette séquence vidéo, HOG a détecté le visage du sujet dans plus de trames que LBP. De plus, HOG a été plus précis en terme d'IoU que LBP.



## 7. Bibliographie

- [1] « Histogram of Oriented Gradients | Learn OpenCV ». [En ligne]. Disponible à: <https://www.learnopencv.com/histogram-of-oriented-gradients/>. [Consulté le: 16-févr-2019].
- [2] « VOT2013 Benchmark | Dataset, Sequence face ». [En ligne]. Disponible à: <http://votchallenge.net/vot2013/dataset.html>. [Consulté le: 10-févr-2019].
- [3] « VOT2013 Benchmark | Dataset, Sequence sunshade ». [En ligne]. Disponible à: <http://votchallenge.net/vot2013/dataset.html>. [Consulté le: 10-févr-2019].
- [4] « VOT2013 Benchmark | Dataset, Sequence trellis ». [En ligne]. Disponible à: <http://votchallenge.net/vot2013/dataset.html>. [Consulté le: 10-févr-2019].
- [5] « VOT2014 Benchmark | Home ». [En ligne]. Disponible à: <http://votchallenge.net/vot2014/>. [Consulté le: 10-févr-2019].
- [6] [En ligne]. Disponible à: [http://dlib.net/face\\_detector.py.html](http://dlib.net/face_detector.py.html). [Consulté le: 20-févr-2019].
- [7] « dlib C++ Library - Image Processing ». [En ligne]. Disponible à: [http://dlib.net/imaging.html#get\\_frontal\\_face\\_detector](http://dlib.net/imaging.html#get_frontal_face_detector). [Consulté le: 20-févr-2019].
- [8] « dlib C++ Library - Image Processing ». [En ligne]. Disponible à: [http://dlib.net/imaging.html#scan\\_fhog\\_pyramid](http://dlib.net/imaging.html#scan_fhog_pyramid). [Consulté le: 20-févr-2019].
- [9] N. Dalal et B. Triggs, « Histograms of Oriented Gradients for Human Detection », (:unav).
- [10] « Cascade Classification — OpenCV 2.4.13.7 documentation ». [En ligne]. Disponible à: [https://docs.opencv.org/2.4/modules/objdetect/doc/cascade\\_classification.html](https://docs.opencv.org/2.4/modules/objdetect/doc/cascade_classification.html). [Consulté le: 20-févr-2019].
- [11] « opencv/lbpcascade\_frontalface.xml at master · opencv/opencv ». [En ligne]. Disponible à: [https://github.com/opencv/opencv/blob/master/data/lbpcascades/lbpcascade\\_frontalface.xml](https://github.com/opencv/opencv/blob/master/data/lbpcascades/lbpcascade_frontalface.xml). [Consulté le: 20-févr-2019].
- [12] « Histogram of Oriented Gradients | Learn OpenCV ». [En ligne]. Disponible à: <https://www.learnopencv.com/histogram-of-oriented-gradients/>. [Consulté le: 26-févr-2019].
- [13] « Local Binary Patterns with Python & OpenCV - PyImageSearch ». [En ligne]. Disponible à: <https://www.pyimagesearch.com/2015/12/07/local-binary-patterns-with-python-opencv/>. [Consulté le: 26-févr-2019].