Cycle d'ingénieur en mathématiques appliquées à modélisation
Vision, Image et Multimédia

Rapport de stage

Recherche d'informations dans des bases de données multimédia naturalistes

Katy Blanc — Laboratoire I3S – équipe MinD

Maîtres de stage
Frédéric Precioso, Diane Lingrand

Tuteur
Lionel Fillatre

Sophia Antipolis, France
1er Septembre 2014

Remerciements

J’aimerais remercier Frédéric Precioso et Diane Lingrand pour leurs conseils, leur professionnalisme, l’atmosphère propice à l’échange et à la discussion instaurée durant ce stage.
J’aimerais également remercier Romaric Pighetti qui m’a aidé pour l’utilisation des serveurs de l’équipe et pour la parallélisation lors du lancement des calculs et Lionel Fillatre pour sa patience et ses conseils permettant l’amélioration de mes qualités de rédaction.

Table des matières

Introduction

La compréhension de séquences d'image est largement utilisée dans l'indexation de vidéos, la programmation de la vision dans la robotique et l'interaction entre les ordinateurs et les humains. Les systèmes de reconnaissance en vision par ordinateur sont généralement décomposés en trois phases : l'extraction de caractéristiques de bas niveau contenues dans la séquence d'image (les formes, les couleurs, les mouvements), la construction d'un descripteur de plus haut niveau à partir de ces caractéristiques permettant l'analyse des données et enfin un système de décision permettant l'interprétation de la scène ou des objets présents.

Le travail présenté dans ce rapport de stage a pour sujet la construction de cette chaîne de traitement pour la reconnaissance de contenu dans des vidéos.

L'équipe MinD du laboratoire I3S concentre leurs activités autour des algorithmes pour l'exploration de données et l'apprentissage appliqués à plusieurs domaines comme le multimédia, la biologie et le texte. L'apprentissage intervient lors de la troisième phase d'un algorithme de reconnaissance pour interpréter la donnée de sorte à l'associer à une classe préalablement apprise ou de sorte à prendre une décision.

Dans le cadre de ma dernière année du cycle ingénieur à Polytech'Nice Sophia en spécialité Vision, Image et Multimédia, j'effectue un stage depuis le 10 mars 2014 pour une durée de 6 mois dans le Laboratoire I3S au sein de l'équipe MinD encadré Frédéric Precioso et Diane Lingrand.

La première partie de mon stage s'est concentrée sur la participation au concours FishClef qui s'est clôturé le 22 mai. Ce concours, créé cette année, consiste à créer un programme permettant de reconnaitre des poissons dans une base de vidéos récupérées grâce à de la vidéo surveillance sous-marine. Mes encadrants avaient participés au concours PlantClef avec Mohamed Issolah, qui consiste en la classification de plantes à partir d'une base de données d'images de feuilles, troncs, fleurs, fruits et plantes entières. Et j'avais moi-même travaillé à l'amélioration de l'algorithme utilisé pour la reconnaissance de plantes dans mon projet de fin d'étude. Ces deux concours font parties du challenge LifeClef.

Dans la seconde partie de mon stage, je me suis intéressée à la description des mouvements dans une vidéo récupérée de la webcam d'un ordinateur. Les mouvements récupérés depuis la séquence d'image est modélisé par un champ de vecteur, le flot optique, qui sera ensuite projeté dans l'espace hilbertien des polynômes pour éliminer les bruits. Une analyse en composante principale (PCA) a été ensuite effectuée pour raccourcir le descripteur du mouvement en conservant la description générale du mouvement. Cette description a ensuite été utilisée pour reconnaitre la posture d'un utilisateur.

Mon rapport se compose donc de deux parties décrivant en détail les deux missions principales de mon stage.

I. Le concours FishClef

1. Description, motivation et évaluation

ImageClef est un concours d'identification de plantes organisé chaque année depuis 2003. Il a vu ainsi la participation de plusieurs groupes de recherches académiques mais aussi de l'industrie provenant de différents domaines comme le data mining, la vision par ordinateur, la reconnaissance de formes, les informations médicales, etc... Comme le nombre de participants a augmenté ces dernières années, le challenge s'est diversifié cette année en proposant un ensemble de trois concours regroupés sous le nom **LifeClef** :

- *PlantClef* cherchant à reconnaître des **plantes** à partir d'une base de données d'**images**.
- *BirdClef* cherchant à reconnaître des espèces d'**oiseaux** à partir d'enregistrements **audio**
- *FishClef* cherchant à reconnaître des espèces de **poissons** à partir d'une base de données de **vidéos**

C'est donc dans ce dernier concours que je me suis investie.

Ces dernières années, des caméras ont été placées dans les fonds marins dans le but de surveiller l'activité sous-marine sans influence humaine. Cependant, il est peu pratique pour des humains d'analyser manuellement la quantité massive de données vidéos quotidiennement produites, car cette tâche exigerait beaucoup de temps et de concentration. Un système automatique d'identification de poissons dans des vidéos est donc d'importance cruciale pour évaluer l'existence et la quantité de poissons présents dans les fonds marins. De plus, il aiderait les biologistes à comprendre l'environnement sous-marin naturel, à promouvoir sa conservation et à étudier les différents comportements et les interactions entre les animaux marins. Le système d'identification automatique devra être robuste face à la faible résolution des vidéos et aux différents phénomènes survenus entre le lever et le coucher du soleil comme des eaux troubles, des algues sur la lentille, etc…

Dans ce but, les organisateurs du concours ont mis à la disposition des participants la base de données vidéos de Fish4Knowledge (www.fish4knowledge.eu) contenant 700k vidéos de 10 minutes filmées au fond du récifs corallien de Taïwan durant les 5 dernières années. La zone de Taïwan est particulièrement intéressante pour étudier l'écosystème marin car elle possède une des plus grandes biodiversités de poissons du monde avec plus de 3000 espèces différentes. Seulement une partie de la base de données (environ 20 k) est fournie et les 10 espèces les plus représentatives du récif ont été annotées pour le concours FishClef.

Le concours FishClef est divisé en 4 tâches :

- identifier les objets en mouvement dans la vidéo
- détecter les poissons dans la vidéo
- détecter les poissons et reconnaître leur espèce dans la vidéo
- reconnaître une espèce de poisson à partir d'une seule image contenant un seul poisson

Notre participation au concours concerne la troisième tâche. Pour cela, nous disposons d'un ensemble d'apprentissage de 285 vidéos avec 19868 poissons avec leur espèce annotés et d'un ensemble de test de 116 vidéos. Il y avait une cinquantaine de participants inscrits pour le concours FishClef et la date de soumission au concours était le 22 mai 2014.

Pour la soumission à la tâche 3, les organisateurs demandaient de fournir un fichier XML contenant pour chaque poisson détecté, sa boîte englobante et sa liste ordonnée d'espèce potentielle, en précisant la vidéo par son id et le numéro de frame.

```
<video id=VIDEO_ID location=LOCATION_CAMERA date_time=DATE width=W height=H frame_rate=FPS gps=GPS>
    <frame number=NUM_FRAME>
        <object id=OBJECT_ID filename=FILE_NAME_OF_THE_IMAGE species=SPECIES_NAME> <!-- SPECIES_NAME is provided for subtask 3 only -->
            <bounding_box>COORDINATES (X,Y) OF BOUNDING BOX </bounding_box> <!-- For subtasks 2 and 3 -->
        </object>
    </frame>
</video>
```

Illustration 1 : Format du fichier XML à fournir lors de la soumission au concours FishClef

Pour mesurer les résultats de reconnaissance, le programme d'évaluation ne prenait en compte uniquement les cadres ayant un score de pascal supérieur à 0.3 par rapport à au moins l'un des cadres des organisateurs.

Soit B_r une bounding-box de référence et B_c la bounding-box courante. Alors le score de Pascal entre ces deux boîtes est calculé par :

$$score\ de\ Pascal = \frac{aire(B_r \cap B_c)}{aire(B_r \cup B_c)}$$

Puis la précision et le rappel pour chacune des espèces sont calculés. Enfin, la précision moyenne et le rappel moyen sur l'ensemble des espèces sont calculés dans le but de comparer les performances des différents algorithmes.

$$Précision\ de\ l'espèce\ i = \frac{|poissons\ correctement\ attribué\ à\ l'espèce\ i|}{|poissons\ classifiés\ de\ de\ l'espèce\ i|}$$

Si la précision est élevée, cela signifie que peu de poissons de l'espèce i ont été classifié comme positive par d'autre espèce. Il y a alors peu de faux positifs. Le système est alors considéré comme « précis ».

$$Rappel\ de\ l'espèce\ i = \frac{|poissons\ correctement\ attribué\ à\ l'espèce\ i|}{|poissons\ de\ l'espèce\ i|}$$

Si le rappel est élevé, alors beaucoup de poisson de l'espèce i ont été classifié par cette espèce. Il y a peu de faux négatifs. Le système est alors considéré comme peu sensible.

En prenant en compte ces critères d'évaluation, nous avons construit une chaîne de traitement capable de détecter, de situer et de reconnaître un poisson dans une vidéo donnée.

2. Système de détection et de reconnaissance de poissons dans des vidéos

Le développement du programme de reconnaissance est entièrement écrit en C++ avec la librairie OpenCV.

1. Étude de l'état de l'art pour la reconnaissance dans les vidéos et tests sur notre base d'apprentissage

Dans le but de classifier le contenu d'une séquence d'images, une étape primordiale est la description des informations caractéristiques présentes.

Se basant sur l'extraction de point d'intérêts d'images ayant déjà montrés leur performance, la méthode STIP (Space-Time Interest Points) est couramment utilisée. Cette méthode permet d'étendre un détecteur conçu pour une image et donc conçu pour un environnement spatial, en 2D, à un environnement spatio-temporel. Dans l'article d'Ivan Laptev [1], le détecteur Harris est étendu de l'image à la vidéo.
L'idée du détecteur de Harris est de détecter les forts changements de I: $\mathbb{R}^2 \rightarrow \mathbb{R}$ dans les deux directions. Ces points d'intérêt sont caractérisés par de fortes valeurs propres de la matrice de second moment μ avec une échelle σ_l^2.

$$\mu(.,\sigma_l^2) = g(.,\sigma_l^2) * \begin{pmatrix} I_x^2 & I_y I_x \\ I_x I_y & I_y^2 \end{pmatrix}$$

où g est un noyau gaussien bidimensionnel de variance σ_l^2 et I_x et I_y représente la dérivé de l'image I selon x et selon y.

$$g(x,y,\sigma_l^2) = \frac{1}{2\pi\sigma_l^2}\exp\left(-\frac{x^2+y^2}{2\sigma_l^2}\right)$$

Ces points d'intérêt sont alors détectés en calculant les maxima positives de la fonction

$$H = \det(\mu) - k.trace^2(\mu)$$

Le paramètre k permet de régler la sensibilité du détecteur de Harris. Il est en général initialiser à 0.04.
Pour étendre cette méthode à un descripteur spatio-temporel, il faut considérer une vidéo comme une fonction I: $\mathbb{R}^2 \times \mathbb{R} \rightarrow \mathbb{R}$ et calculer sa matrice de second moment

$$\mu = g(.,\sigma_l^2,\tau_l^2) * \begin{pmatrix} I_x^2 & I_x I_y & I_x I_t \\ I_x I_y & I_y^2 & I_y I_t \\ I_x I_t & I_y I_t & I_t^2 \end{pmatrix}$$

où g est un noyau gaussien spatio-temporel définit par

$$g(x,x,t,\sigma_l^2,\tau_l^2) = \frac{1}{\sqrt{(2\pi)^3\sigma_l^4\tau_l^2}} \times \exp\left(-\frac{x^2+y^2}{2\sigma_l^2} - \frac{t^2}{2\tau_l^2}\right)$$

Et $I_x, I_y et\ I_t$ sont les dérivées de la séquence d'image selon les différents axes.
Il est important que le noyau gaussien ai une variance différents en échelle et en temps cat l'étendu spatiale et temporel d'un évènement sont en général indépendant.
Les points d'intérêt sont alors détectés grâce aux maxima locale positive de

$$H = \det(\mu) - k.trace^3(\mu)$$

Comme en spatial, k permet de régler la sensibilité du détecteur. Il est en général initialisé à 0.005.

L'étude [2] détaille et compare les détecteurs et descripteurs 3D les plus connus pour la reconnaissance d'action. Les détecteurs étudiés sont Harris 3D (le STIP basé sur le détecteur Harris), le détecteur Cuboid basé sur des filtres temporels de Gabor appliqués au voisinage d'un pixel, le détecteur Hessian basé sur la matrice hessienne 3D à différentes échelles dans le voisinage d'un pixel et enfin le détecteur dense.
Les descripteurs comparés sont le descripteur cuboid (concaténation des gradients de chaque pixels dans un cuboïde puis analyse en composantes principales (ACP) de sorte à réduire la dimension), le descripteur HOG/HOF (concaténation des histogrammes d'orientation de gradients spatiaux et des histogrammes d'orientation du flot optique), le descripteur HOG3D (un histogramme d'orientation de gradient en 3D) et le

descripteur étendu SURF (STIP basé sur le descripteur 2D SURF). Nous détaillerons les descripteurs les plus complexes ci-dessous.

Le descripteur HOG se base sur le calcul des orientations des gradients d'une image.
Le gradient d'un pixel d'une image f peut être calculé de différente façon de sorte à approcher la forme exacte $\nabla f = \frac{\delta f}{\delta x}x + \frac{\delta f}{\delta y}y$. Une méthode consiste à convoluer l'image par le filtre de Sobel $\begin{pmatrix} 1 & 0 & -1 \\ 2 & 0 & -2 \\ 1 & 0 & -1 \end{pmatrix}$ pour obtenir les dérivées horizontales G_x en chaque point et par $\begin{pmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & 2 & -1 \end{pmatrix}$ pour obtenir les dérivées verticales G_y. Ainsi l'image $G = \sqrt{G_x^2 + G_y^2}$ contient les gradients de chaque pixel. L'image fournit par $\theta = \text{atan}\left(\frac{G_y}{G_x}\right)$ fournit les orientations des gradients en chaque pixel.
Ayant maintenant l'orientation et l'intensité du gradient en chaque pixel, on découpe une zone autour du point clés à décrire en carrés de petite taille (de 4x4 à 12x12 pixels) et on crée un histogramme des orientations en pondérant chaque vote par l'intensité du gradient à ce pixel. L'histogramme est ensuite normalisé pour éviter les disparités dues aux changements d'illumination.

Le HOF est construit en considérant le même voisinage du point d'intérêt en effectuant un histogramme sur les orientations du flot optique extrait. L'histogramme est obtenu sur 5 bins décrivant les mouvement vers la droite, la gauche, le haut, le bas et aucun mouvement.
Le calcul du flot optique est basé sur l'hypothèse de conservation de la luminance, c'est-à-dire que la luminance d'un point physique de l'image de varie pas au cours du temps.

$$I(x, y, t) = I(x + dx, y + dy, t + 1)$$

Les techniques de détermination du flot optique permettent de résoudre l'équation de conservation du mouvement apparent :

$$(\nabla I)^t v + I_t = 0 \qquad (ECMA)$$

Où I est l'intensité de la frame et v le vecteur vitesse du pixel (x,y) au temps t à déterminer.
Un algorithme efficace et robuste pour le calcul du flot optique a été conçu par Gunnar Farneback [4]. La première étape de cet algorithme est d'approximer chaque voisinage de pixel dans les deux frames successives par un polynôme quadratique.

$$f(x) = x^T A x + b^T x + c$$

Où A est une matrice symétrique, b un vecteur et c un scalaire.
Puis on estime le déplacement v en observant comment une translation de vecteur v transforme les coefficients du polynôme

$$f_2(x) = f_1(x - v) = (x - d)^T A_1 (x - d) + b_1^T (x - d) + c_1 = x^T A_2 x + b_2^T x + c_2$$

Les relations suivantes sont alors obtenues :

$$A_2 = A_1$$
$$b_2 = b_1 - 2A_1 v$$
$$c_2 = v^T A_1 v - b_1^T v + c_1$$

Si la matrice A_1 est inversible, alors de la deuxième équation nous obtenons $v = \frac{-1}{2} A_1^{-1}(b_2 - b_1)$.

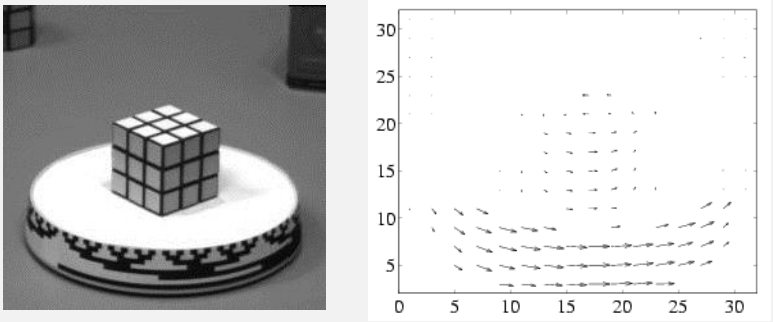

Illustration 2 : Flot optique d'un objet en rotation

Le descripteur HOG/HOF est obtenu en concaténant les histogrammes HOG et HOF pour ainsi décrire les formes et les mouvements.

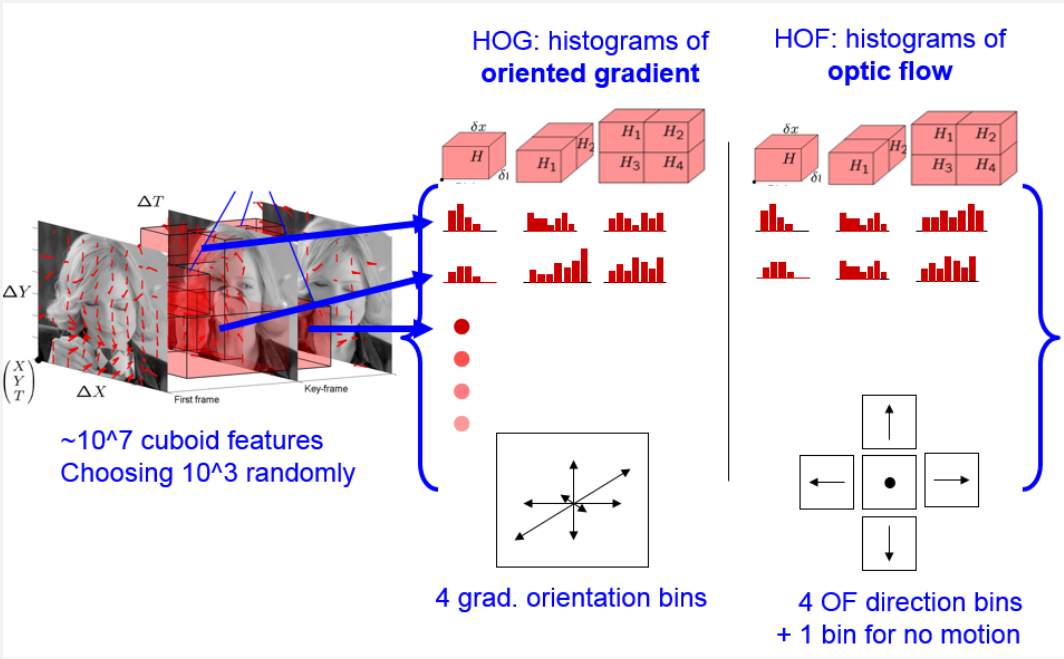

Illustration 3: Les descripteurs HOG et HOF

Le descripteur HOG3 est obtenu de la même façon que le descripteur HOG mais en calculant un gradient spatio-temporel au lieu d'un descripteur spatial.

Le descripteur SURF est un descripteur basé sur le descripteur SIFT (qui sera détaillé par la suite) dont le calcul est trois fois plus rapide mais fournit de moins bonne performance. Cette méthode calcule les détecteur SIFT sur l'image intégrale au lieu de l'image de base. L'image intégrale est obtenue par

$$Iint(x,y) = \sum_{i \le x\,; j \le y} I(i,j)$$

Cette étude montre que les descripteurs HOG/HOF et HOG3D fournissent les meilleurs résultats pour la

reconnaissance d'actions, associés aux détecteurs dense ou cuboïde.

De sorte à extraire le maximum de caractéristiques d'un objet en mouvement dans une vidéo, il possible de suivre l'objet à partir d'une position initiale.

La méthode de tracking CamShift permet de suivre un objet en mouvement en se basant sur son histogramme (généralement dans l'espèce de couleurs HSV). A un instant t, une zone de référence à suivre est fournie et, à partir de son histogramme, une backprojection de l'image à un temps t+1 est construite. Cette backprojection est construite en associant à chaque pixel la valeur de l'histogramme de l'image précédente. Cette valeur est alors comme une probabilité du pixel d'appartenir ou pas à la zone recherchée. La zone maximale de cette image est alors repérée par la méthode Mean-Shift, qui permet de détecter des optima sur une surface, et est désignée comme la zone à trouver. Cette méthode peut être utilisé avec différents espaces de couleurs et en choisissant un ou plusieurs canaux de couleurs.

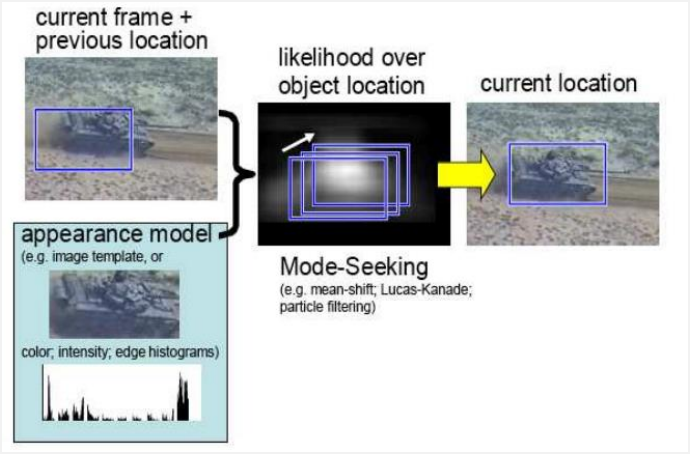

Illustration 4 : Schéma de fonctionnement de la méthode CamShift

La méthode CamShift a été améliorée dans l'article [3] en l'associant à une segmentation de la zone à suivre et en suivant séparément chaque zone de couleur dominante puis en combinant les backprojections obtenues. Nous avons testé cette méthode sur la base de données du concours FishClef.

Illustration 5 : Tracking par la méthode CamShift
La fenêtre bleue représente le cadre depuis lequel on part pour traquer le poisson dans la frame à un temps t et sur l'image de droite, la frame au temps t+1, le fenêtre rouge représente la zone de recherche et la fenêtre jaune la zone sélectionnées comme correspondante

Comme on peut le constater sur l'illustration 5, cette méthode ne donna pas de bons résultats. Cela est dû essentiellement aux changements de luminosité dans la vidéo et du fait que certaines algues possèdent des couleurs assez proches des poissons à repérer.

Une autre méthode de tracking testée a été la segmentation du flot optique. Le calcul du flot optique

est une technique permettant d'estimer le mouvement à partir de deux images successives en construisant un vecteur de mouvement par pixel ce qui fournit alors un champ de vecteurs vitesse. Nous avons utilisé la technique de Gunnar Farneback qui a été expliqué précédemment.
Sur notre base de vidéos, l'estimation du flot optique n'était pas performante car l'hypothèse de conservation de la luminance n'est pas respectée.

Le tracking par descripteurs est également une méthode fortement utilisée dans le suivi d'un objet dans une vidéo. Cela consiste à extraire des points clés grâce à des détecteurs dans deux frames successives et à les décrire, puis à effectuer une association entre les points de chacune des images de sorte à retrouver la zone ayant des points clés similaires à la zone d'intérêt. Cette association est effectuée par un calcul du plus proche voisin d'un point de la première frame dans la seconde. Nous avons également calculé une homographie, c'est-à-dire une transformation géométrique d'un ensemble de points clés à l'autre à partir des associations, de sorte à retrouver une même disposition des points clés détectés. Mais comme notre poisson ne se trouve pas souvent dans la même position d'une frame à l'autre, la recherche d'homographie n'était pas pertinente. Le tracking par descripteur donnait des résultats plutôt corrects comme vous pouvez le voir sur l'illustration 6.

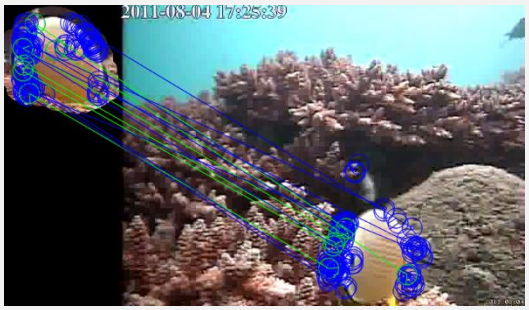

Illustration 6 : tracking par descripteur

Nous avons utilisé l'association ayant la plus petite distance entre les deux points clés pour pouvoir construire un encadrement autour du poisson et ainsi le traquer. Cette méthode se trouve être performante en choisissant le détecteur et le descripteur adéquat, nous avons choisi le détecteur Surf et le descripteur Opponent Sift que je vous présenterai par la suite. Mais les coraux était quelques fois confondu avec les poissons, car ces derniers ont des formes assez diverses pouvant se rapprocher de la forme du poisson.
Le tracking insérré dans la méthode de reconnaissance doit permettre de suivre le poisson parfaitement de sorte à extraire plus de descripteurs pour cette espèce. Ce suivi doit être particulièrement discriminant quitte à manquer quelques fois le poisson, de sorte à ne pas insérer dans l'apprentissage des descripteurs qui ne se situe pas sur le poisson et ainsi à éviter les faux positifs. Pour cette raison, nous avons décidé de ne pas insérer le tracking dans notre chaîne de traitement.
Forts de ces études et expériences, nous avons construit un système de reconnaissance décomposé en deux parties :

- La partie off-line : nous exploitons les metadata de la base de données et la base d'apprentissage pour préparer le système de reconnaissance
- La partie on-line : lors de l'acquisition de la vidéo à traiter, nous détectons, encadrons et déterminons l'espèce des poissons présents.

2. Partie off-line du système : récupération des caractéristiques de chacune des espèces

Les organisateurs ont fournis avec les vidéos de la base un fichier XML du même format que celui imposé détaillant l'emplacement de poissons en précisant la vidéo, la frame et le rectangle encadrant plusieurs poissons en précisant leur espèce. En se basant sur ces metadata, nous avons extraits des points clés sur chacun des poissons annotés. Après avoir essayé plusieurs détecteurs de points clés dans les images comme SIFT, SURF et Harris, nous avons remarqué les points clés fournis n'avaient pas une assez grande échelle pour pouvoir décrire les formes et les couleurs du poisson. En effet, comme les vidéos de la base sont très pixélisées, nous avons adopté la stratégie basée sur des points clés avec de larges échelles pour avoir des descriptions plus globales du poisson. C'est pourquoi nous avons construit trois points clés à différentes

échelles, situé sur l'axe horizontal à $\frac{1}{3}, \frac{1}{2} et \frac{2}{3}$ de la longueur horizontale. Les échelles sont initialisées à la taille de la boite, puis augmente et diminue légèrement (de 10 en 10 pixels) de sorte à contenir entièrement le poisson ainsi que ses principales parties comme sa tête, son corps et sa queue.

Illustration 7 : points clés de larges échelles décrivant chaque poisson de la base

Puis nous avons décrit ces points clés en utilisant le descripteur Opponent Sift.
Le descripteur Sift [5] est construit à partir d'un point clés en connaissance son échelle et son orientation. On extrait le voisinage de ce point grâce à ses informations et on calcul le gradient de l'image en chaque pixel avec sa magnitude et son orientation (comme expliqué dans la section I.2.1).
Dans le but que le descripteur soit invariant aux rotations, l'image extraite du voisinage subit une rotation dans le sens opposé à l'orientation du point clé. Puisque l'on a la magnitude et l'orientation en chaque pixel, le voisinage du point clés est découpé en 16 parties (4x4), découpé elles-mêmes en 16 mini-parties (4x4) dont chaque mini-partie sera représentée par une magnitude et une orientation. Pour chaque partie, un histogramme des orientations sur 8 directions est construit. Enfin ces 16 histogrammes sont concaténés de sorte à composer le descripteur Sift.

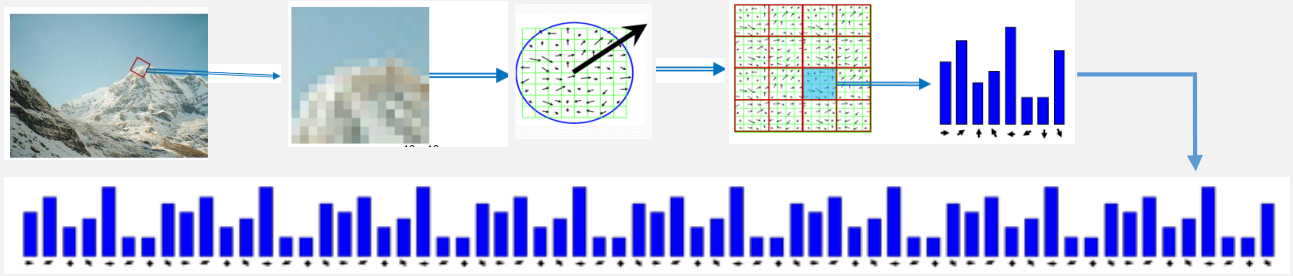

Illustration 8 : Le descripteur SIFT de taille 128

Basé sur le gradient d'une image en niveau de gris, ce descripteur caractérise les formes autours du point clés en étant invariant aux changements de luminosité, d'orientation et de point de vue. Ce descripteur a été étendu [6] dans le but de décrire également les couleurs. Pour ce faire, les pixels sont exprimés dans l'opponent color space.

$$\begin{pmatrix} O_1 \\ O_2 \\ O_3 \end{pmatrix} = \begin{pmatrix} \frac{R-G}{\sqrt{2}} \\ \frac{R+G-2B}{\sqrt{6}} \\ \frac{R+G+B}{\sqrt{3}} \end{pmatrix}$$

Puis le descripteur SIFT est calculé sur chaque canal de couleur : l'Opponent Sift descriptor est la concaténation de ces descripteurs SIFT et du détecteur SIFT calculé sur les niveaux de gris de l'image. Ainsi le descripteur Opponent Sift a une taille de 512.
Ce descripteur possède l'avantage de décrire la forme mais également les couleurs qui sont caractéristiques chez les espèces de poissons.
La partie off-line du système consiste donc à extraire et stocker les descripteurs de chacune des espèces.

3. Partie on-line : détection et reconnaissance

a. *Détection de l'arrière-plan*

Dès l'acquisition de la vidéo à analyser, la première phase du traitement est de dissocier l'arrière-plan des poissons. Le but de cette étape est de diminuer les zones de la vidéo à traiter et ainsi améliorer la performance ainsi que la complexité de l'algorithme. Pour cela nous avons utilisé la technique de segmentation d'arrière-plan par mixture de gaussiennes [7]. Cette méthode consiste à attribuer à un pixel une loi de probabilité sur ces valeurs possibles grâce aux T enregistrements précédents X_T . Cette probabilité s'exprime sous forme d'une somme pondérée par π_m de gaussiennes de moyennes μ_m et de variances σ_m qui sont calculées récursivement.

$$\hat{p}(\vec{x}|\mathcal{X}_T, BG+FG) = \sum_{m=1}^{M} \hat{\pi}_m \mathcal{N}(\vec{x}; \widehat{\vec{\mu}}_m, \widehat{\sigma}_m^2 I)$$

Avec les équations de récurrence suivantes :

$$\begin{aligned}
\hat{\pi}_m &\leftarrow \hat{\pi}_m + \alpha(o_m^{(t)} - \hat{\pi}_m) \\
\widehat{\vec{\mu}}_m &\leftarrow \widehat{\vec{\mu}}_m + o_m^{(t)}(\alpha/\hat{\pi}_m)\vec{\delta}_m \\
\widehat{\sigma}_m^2 &\leftarrow \widehat{\sigma}_m^2 + o_m^{(t)}(\alpha/\hat{\pi}_m)(\vec{\delta}_m^T\vec{\delta}_m - \widehat{\sigma}_m^2)
\end{aligned}$$

Où $\vec{\delta}_m = \vec{x}^{(t)} - \widehat{\vec{\mu}}_m$, $\alpha = 1/T$ et $o_m^{(t)}$ est mis à 1 pour les composants proches selon la distance de Mahalanobis $D_m^2(\vec{x}^{(t)}) = \vec{\delta}_m^T\vec{\delta}_m/\hat{\sigma}_m^2$ avec les plus fort poids et 0 pour les autres.
S'il n'y a pas de composantes proches, alors un nouvel élément est introduit dans la somme avec $\hat{\pi}_{M+1} = \alpha$, $\widehat{\vec{\mu}}_{M+1} = \vec{x}^{(t)}$ et $\hat{\sigma}_{M+1} = \sigma_0$, une variance initiale choisie.
Si la somme est constituée de plus de M termes, la loi avec le plus petit poids π_m est supprimée. Ainsi le premier plan sera représenté dans la valeur des pixels par les plus petits poids π_m et le background peut être modélisé par les gaussiennes de plus fort poids. Plus précisément, l'arrière-plan est modélisé par les lois ayant les plus fort poids

$$p(\vec{x}|\mathcal{X}_T, BG) \sim \sum_{m=1}^{B} \hat{\pi}_m \mathcal{N}(\vec{x}; \widehat{\vec{\mu}}_m, \sigma_m^2 I)$$

Avec

$$B = \arg\min_b \left(\sum_{m=1}^{b} \hat{\pi}_m > (1 - c_f) \right)$$

Et c_f est une mesure choisie de la portion maximale d'information appartenant au premier plan.
Comme les poissons sont constamment en mouvement contrairement aux algues qui effectue des mouvements oscillatoires aux mêmes endroits, ce modèle fournit de très bonnes performances pour la segmentation de l'arrière-plan.
De sorte à éliminer les bruits et à englober entièrement le poisson dans le masque de mouvement, nous avons ensuite effectué une érosion circulaire de rayon 3 puis une dilatation circulaire de rayon 10 sur ce masque.

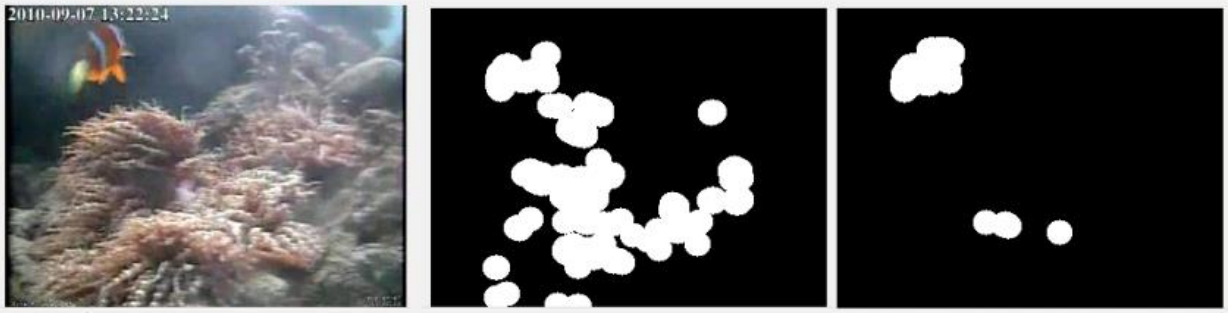

Illustration 9 : Efficacité du modèle des mixtures de gaussiennes dans la segmentation du background
De gauche à droite, frame d'une vidéo, masque de mouvement à partir d'une différence entre la frame et une moyenne des frames sur la vidéo entière et la masque de mouvement obtenu grâce au modèle des mixtures de gaussiennes

Nous pouvons voir nettement sur l'illustration ci-dessus que, sur cette vidéo où il y a beaucoup d'algues en mouvements, la détection par mixture de gaussiennes est plus efficace qu'une simple différence avec la moyenne de frame de la vidéo puisqu'elle permet de détecter le poisson mais pas les algues en mouvement.
Maintenant que nous avons un masque permettant de filtrer les zones contenant des poissons, nous devons les décrire et les comparer aux descripteurs extraits durant la phase off-line.

b. *Description, apprentissage et reconnaissance*

Une chaîne de traitement classique en reconnaissance consiste à effectuer directement l'apprentissage à partir des descripteurs extraits de la base d'apprentissage. Grâce à notre masque de mouvement, nous avons l'arrière-plan dans lequel le poisson apparaîtra. Nous avons donc décidé d'exploiter cette donnée de sorte à décrire le background avec le descripteur Opponent Sift et l'opposé aux descripteurs des espèces. Ainsi, même si certaines zones du masque se trouvent sur l'arrière-plan, le classifieur préalablement entrainé reconnaîtra ces descripteurs comme appartenant au background.

Pour extraire des points clés du background, nous avons choisi le détecteur dense comparé aux autres détecteurs pour les raisons suivantes:

- toutes les zones du background seront ainsi décrites
- les points clés sont assez nombreux pour apporter une quantité d'informations suffisantes
- les échelles peuvent être initialisées de sorte à avoir des descripteurs de même échelles que lors de la description des espèces

De plus, comme montré dans l'étude comparative [2], le détecteur dense fournit de meilleures performances sur les vidéos.
Nous avons donc décrit le background grâce aux zones sombres du masque dans le début des vidéos jusqu'à avoir autant de descripteurs que les descripteurs des espèces. Plus précisément, les points clés sont distribués de façons régulièrement dans les zones sombres du masque avec une échelle allant de 30 à 110 pixels pour le diamètre des points clés.

Lors de la description des espèces, il est possible que l'échelle des points clés fût si importante qu'une partie du fond a été décrite au lieu de l'espèce à reconnaître. Donc, avant d'apprendre à classifier ces descripteurs, nous avons effectué un filtrage des descripteurs des espèces. Pour chaque point clé de l'espèce, nous étudions ses 10 plus proches voisins dans la liste des descripteurs constituée de ceux du background et ceux de l'espèce courante. Les meilleurs descripteurs de l'espèce seront ceux dont la description possède plus de voisins dans l'espèce que dans ceux de l'arrière-plan et les meilleurs descripteurs seront conservés jusqu'à atteindre une nouvelle liste de descripteurs constituée d'au moins la moitié des descripteurs initiaux.

Après avoir effectué ce filtrage, nous pouvons apprendre à reconnaître les espèces. La technique d'apprentissage par Support Vector Machine (SVM) fut utilisée dans cette phase. Cette technique consiste à trouver l'hyperplan séparateur des deux classes de descripteurs dans l'espace vectoriel de même dimension de sorte à trouver la marge maximale séparant ces deux classes.

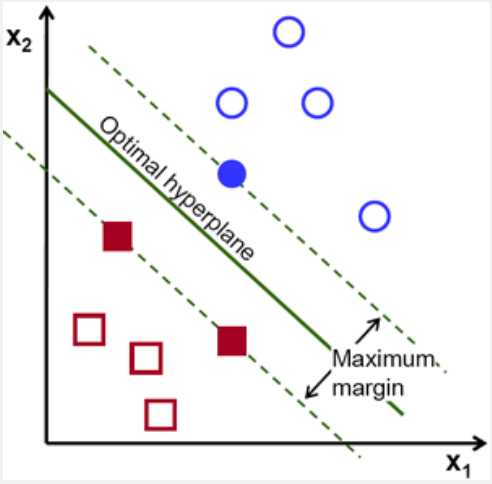

Illustration 10 : Représentation d'un système d'apprentissage par Support Vector Machine

Dans la phase d'apprentissage, un SVM sera entraîné à reconnaître chaque classe contre l'arrière-plan de la vidéo. Une fois chaque SVM entraîné, un descripteur sera classé comme appartenant à une espèce si le SVM correspondant classe ce descripteur dans la classe des descripteurs de l'espèce bien entendu mais avec la condition supplémentaire que ce descripteur soit à une distance de 1 de l'hyperplan séparateur, c'est-à-dire que ce descripteur ne se situe pas dans la marge. Plus précisément, un score est attribué à chaque descripteur pour chaque SVM : la distance à l'hyperplan séparateur signé par la classe dans laquelle il se situe (-1 pour le background et 1 pour l'espèce). Si le descripteur à un score supérieur à 1, il est considéré comme appartenant à l'espèce. Si le descripteur répond à ce critère pour plusieurs espèces, les espèces seront ordonnées par score. Ainsi le meilleure score sera obtenu par le SVM de l'espèce dans lequel le descripteur est le plus éloigné de la marge et donc de l'arrière-plan.

Maintenant que nous avons les classifieurs, nous avons effectué la première ligne du traitement on-line comme vous pouvez le voir sur le schéma ci-dessous.

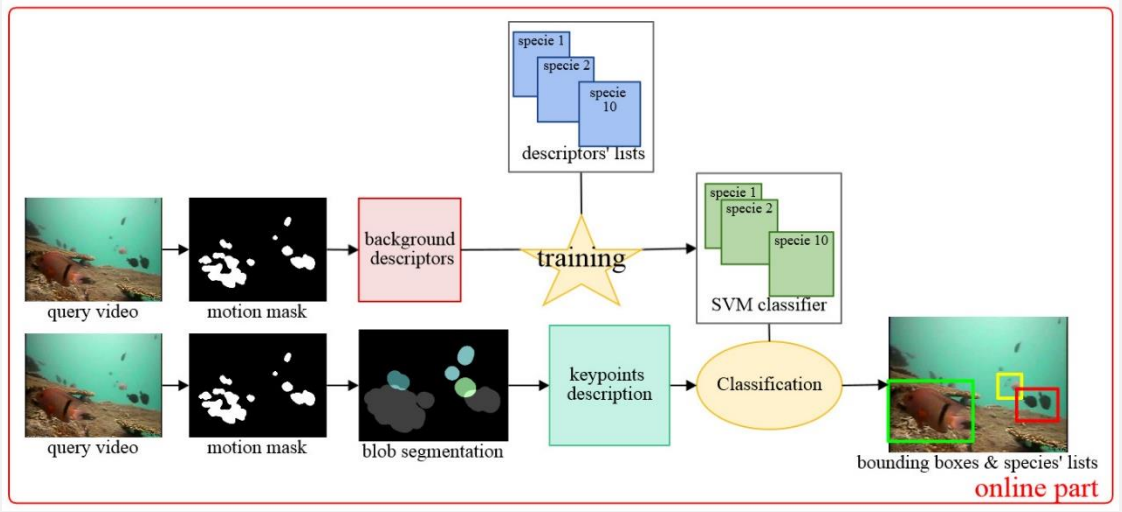

Illustration 11 : Partie on-line de la chaîne de reconnaissance

c. Construction des boîtes englobantes et soumission

Tout d'abord, nous appliquons le masque de mouvement pour réduire la zone de recherche. Puis chaque « tâche » du masque de mouvement est considérée comme un poisson. Alors sur chacun de ces « blobs », nous extrayons densément des points clés avec différentes échelles, les mêmes échelles utilisées lors de la description du background (de 30 à 110). Puis chaque point clés est décrit par Opponent Sift et chaque classifieur leur attribue un score comme décrit précédemment. Les scores en dessous de 1 ne sont pas pris en compte. Puis, par blob, les scores des descripteurs sont additionnés par espèce. Ainsi l'espèce ayant obtenue la meilleure somme à l'intérieur du blob, et donc ayant le plus de descripteurs de cette espèce, sera le premier choix d'espèce, la deuxième meilleure somme la seconde espèce, etc... Les espèces dont la somme des scores vaut 0 ne sont pas insérrées dans la liste d'espèces potentielles.

Les blobs sont également filtrés en sélectionnant uniquement ceux qui ont plus de deux points clés détectés et qui ont une proportion de points clés reconnus (score au-dessus de 1) supérieure à 20 % pour éliminer les cas où le nombre de points clés détectés est tellement important que quelques-uns sont classés positivement, par exemple sur les coraux qui ont des formes assez aléatoires et lors d'un changement de luminosité assez important pour troubler le masque de mouvement.
Donc les blobs, qui peuvent être également considérés comme des ensembles de descripteurs, possèdent chacun leur liste ordonnée d'espèces potentielles.

Pour la construction de la boîte englobante, nous avons choisi d'encadrer entièrement tous les points clés (avec leur diamètre de voisinage) classifiés comme appartenant à l'espèce placée en première position du blob.

La liste des bounding box obtenues situées dans les vidéos par le numéro de frame et associées à leur liste ordonnée d'espèces potentielles est rédigée dans un fichier XML au format demandé par les organisateurs du concours. Ce fichier XML constitue alors notre premier run pour la soumission au concours FishClef.

Les améliorations suivantes apportées au système de reconnaissance consistent à séparer les poissons à l'intérieur d'un même blob. Le choix de diminuer la taille des bounding-box par la diminution de la taille des blobs est nécessaire pour que ces boîtes respectent le critère du score de Pascal et ainsi qu'elles soient prises en compte. En effet, comme les groupes de points clés étaient séparés par blob, deux poissons se trouvant dans le même blob étaient classés comme un gros poisson et ce cas se produisait souvent puisque la dilatation des blob devait être assez importante pour ne pas perdre d'information et surtout pour ne pas diviser un poisson en deux blobs non connexes. Pour pallier à ce problème, une dilatation moins grossière, de rayon 3, a été effectuée et chaque « mini blob » dont la couleur dominante était la même et qui étaient assez proches spatialement ont été associés comme étant du même poisson.

La couleur dominante a été désignée approximativement en effectuant un histogramme en trois dimensions sur l'espace couleur RGB avec un découpage de chacun des canaux en 3 bins. Cela nous fait donc 27 couleurs dominantes possibles ce qui est tout à fait satisfaisant pour différentier les poissons. D'autant plus que cette distinction ne doit pas être trop fine pour ne pas classer séparément deux parties d'un même poisson, par exemple la queue du poisson pourrait être légèrement plus assombrie que son corps. Et deux blobs ont été désignés comme « spatialement proche » si ils sont considérés comme appartenant au même blob avec une dilatation de rayon 10.

Enfin une autre dilatation légère est effectuée sur chaque blob séparément. Ainsi un point clé peut appartenir à plusieurs blobs mais il n'y a pas de pertes d'information et les poissons proches ne sont pas classifiés ensemble.

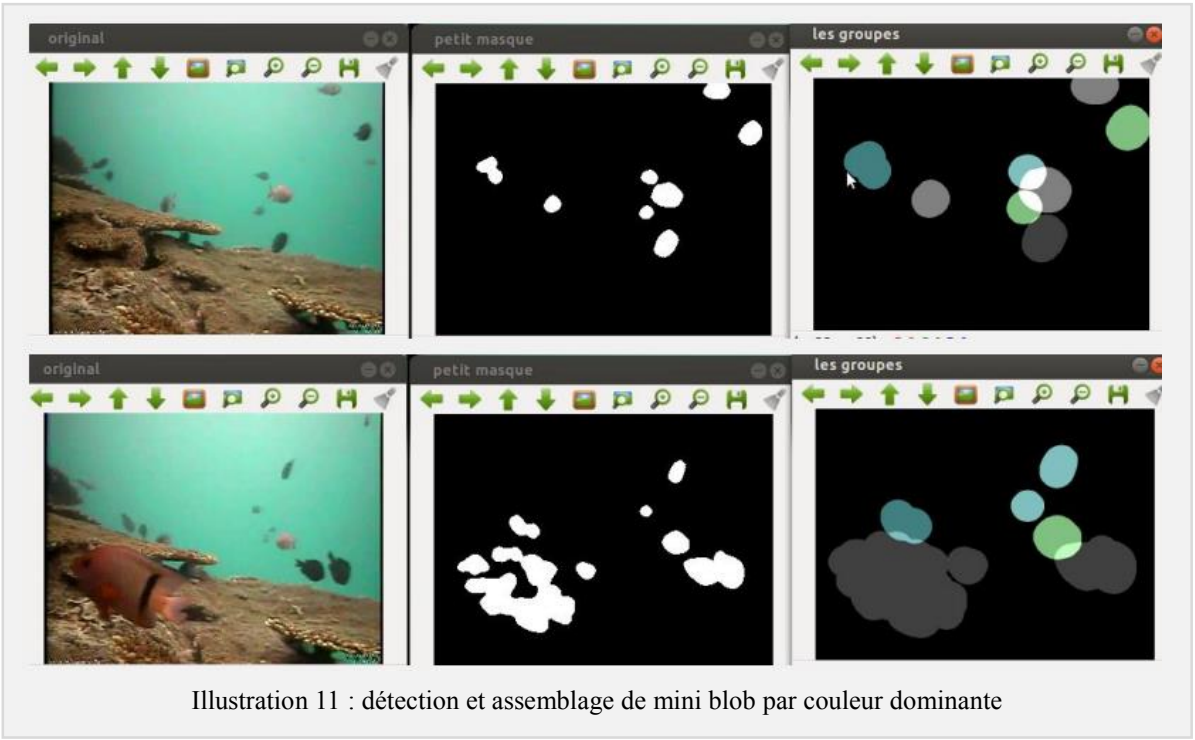

Illustration 11 : détection et assemblage de mini blob par couleur dominante

On peut voir dans l'illustration ci-dessus que les trois poissons très proches dans la première frame sont bien détectés et ne sont pas associés (car il y a une intersection d'une couleur différente de leur propre couleur), et dans la deuxième frame, les minis blobs du poisson en bas à gauche sont bien représentés tous dans un même blob.

Grâce à cette nouvelle segmentation du masque de mouvement, nous effectuons l'extraction des descripteurs et l'association de chaque blob à une liste d'espèce et à une bounding-box selon les mêmes critères que précédemment.

Pour la participation au concours FishClef le jeudi 22 mai, nous avons soumis la chaîne de traitement présentée ci-dessus avec 3 sorties différentes :

- une avec les bounding-box par tâches sur le masque de mouvement
- une avec les différentiations/associations de minis tâches par couleur dominante
- une avec chacune des minis tâches classées séparément, puis les tâches obtenues par association par couleur dominante

3. Résultats du concours et publications

Sur les cinquante participants inscrits au concours FishClef, seulement deux équipes ont soumis des résultats : une équipe pour la tâche 4 de reconnaissance à partir d'images et notre équipe pour la tâche 3 de détection et de reconnaissance. Les organisateurs ont construits un programme « baseline » pour pouvoir comparer nos résultats aux leurs. Leur baseline est constituée de l'approche de modélisation du background ViBe[8] pour la détection des poissons dans la vidéo et de la chaîne de traitement VLFeat+Bag of Word [9] pour la reconnaissance des espèces. Les scores de précision et de rappel pour les trois soumissions sont détaillés dans les graphes suivants.

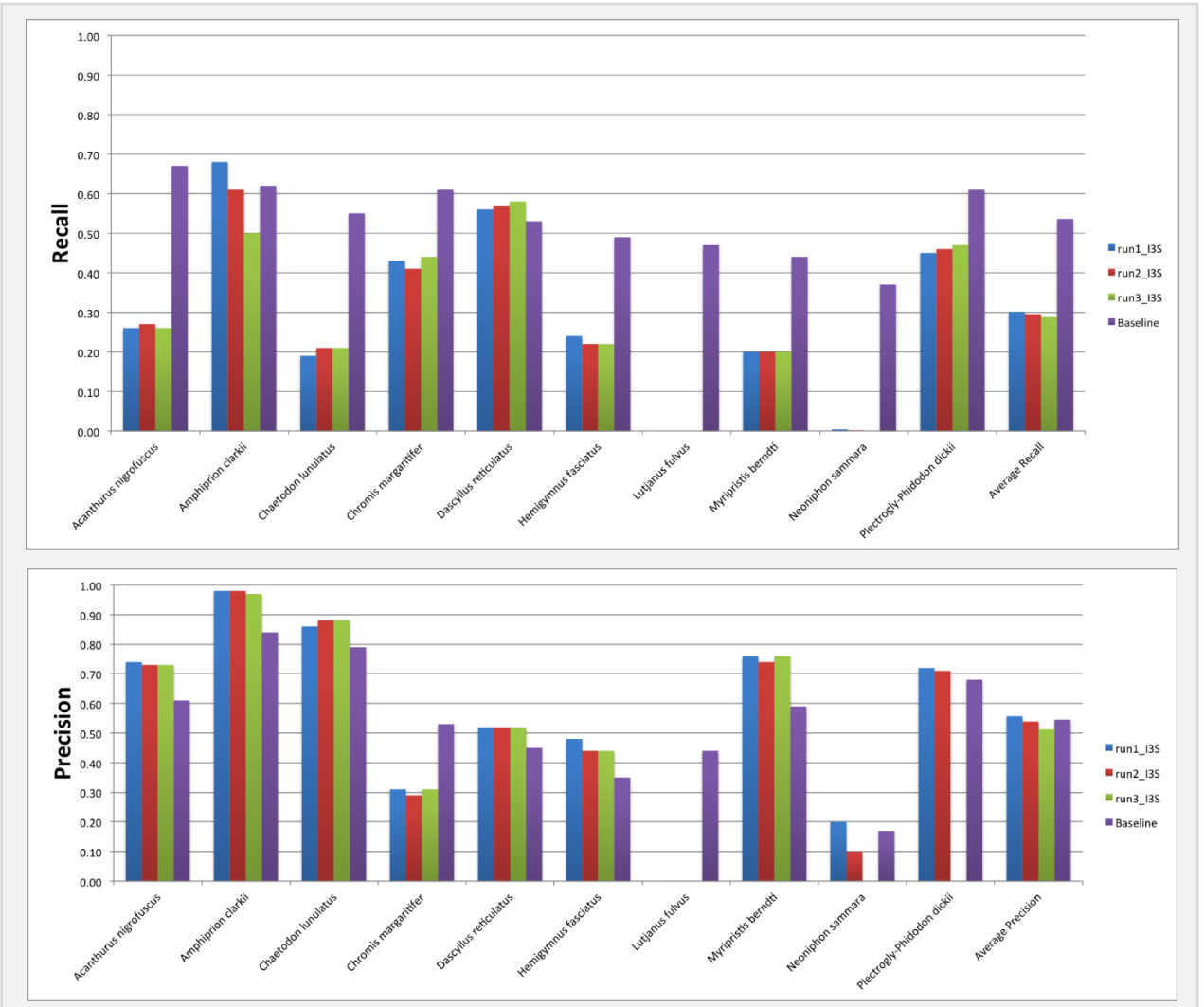

Illustration 12 : Scores de précision et de rappel calculés par espèce et en moyenne pour chacune des soumissions.

On peut remarquer tout d'abord que nos scores de rappel ne sont pas très élevés. Cela est dû au fait que nos bounding-box étaient souvent trop grosses par rapport à celles des organisateurs pour être prises en compte (critère du score de Pascal). En effet, nous avons englobé tous les points clés mais également leur échelle lors de la construction des boîtes. Pour obtenir des bounding-box plus fines, il faudrait englober uniquement les points clés.
Comparé à leur baseline, nous avons donc une meilleure précision mais un moins bon rappel. Donc notre programme de reconnaissance manque plusieurs poissons, mais lorsqu'il est détecté, la bonne espèce est fournie.

Cette conclusion se confirme par l'application, ultérieure au concours, de notre programme de reconnaissance à la tâche 4, c'est à dire sans la détection des poissons. Plus précisément, cette tâche consiste à reconnaître un poisson depuis une image de petite taille ne contenant qu'un poisson. Pour adapter notre programme, nous avons construit nos groupes de points clés de la même manière que lors de la phase off-line du système : trois groupes de point clés à différentes échelles sur l'axe horizontal de l'image. Nous avons ensuite décrits ces points clés par l'Opponent Sift et classifier ces descripteurs par nos SVM déjà entraînés. Nos résultats sont détaillés dans le tableau ci-dessous.

Fish Species	Image-based approach	Video-based approach	Baseline
Acanthurus nigrofoscus	0.96	0.85	0.99
Amphiprion clarkii	1	0.88	0.90
Chaetodon lunulatus	1	0.91	0.92
Chromis margaritifer	1	0.99	0.90
Dascyllus reticulatus	0.98	0.94	0.88
Hemygimnus fasciatus	0.99	0.92	0.92
Lutjanus fulvus	1	0.91	0.93
Myripristis berndti	0.97	0.98	0.89
Neoniphon sammara	1	0.98	0.92
Plectrogly-Phidodon dickii	0.99	0.98	0.89
Average	0.99	0.93	0.91

Illustration 13 : Scores de rappel calculés par espèce et en moyenne

Le terme image-based approach fait référence à la soumission de l'équipe DYNI83 à la tâche 4, le terme video-based approach à l'adaptation de notre algorithme initialement conçu pour les vidéos et la baseline à l'algorithme des organisateurs.

Notre chaîne de traitement pour la détection et la reconnaissance dans les vidéos a été décrite dans une note de travail qui sera bientôt disponible sur le site du concours ImageClef.

Nous avons également publié un article co-écrit avec un des organisateurs du concours, C. Spampinato, et l'équipe DYNI83, P.H. Joalland, S. Paris et H. Glotin, dans le journal Multimedia Tools and Applications (MTAP).

Le 15 Septembre, j'aurais le privilège de présenter notre travail et la participation de Mohamed Issolah de l'équipe Mind au concours PlantClef à la conférence CLEF2014 à Sheffield et d'y exposer un poster résumant notre chaîne de traitement pour la reconnaissance des poissons.

Enfin, notre programme de détection et de reconnaissance a été sélectionné pour être présenté à la conférence 3rd ACM international regular and data challenge workshop on multimedia analysis for ecological data (MAED 2014) à Orlondo le 7 Novembre 2014.

II. Description de contenu vidéo grâce à la projection de flot optique

1. Étude bibliographique de description de l'information temporelle

L'avantage principal d'effectuer de la reconnaissance dans des vidéos par rapport aux images est l'exploitation des mouvements apparaissant au cours de la séquence. Le mouvement constitue une source d'information visuelle primordiale dans la reconnaissance d'actions, de phénomènes naturelles et de comportements de foules par exemple.

Une nouvelle méthode original pour caractériser l'information temporelle a immergée dernière. De l'idée que nous percevons l'information temporelle avec des variations plus lentes que les capteurs a été créé la Slow Features Analysis [13,14]. Cette méthode consiste à partir d'un échantillon de vecteurs variant au cours du temps, et d'en extraire de l'information à variations lentes, de moindre dimension et en créant des signaux décorrélés appelé slow features.

Soit $x(t) = [x_1(t), \dots, x_I(t)]^T$ un signal d'entrée de dimension I. Les slow features de x sont représentés dans un signal de sortie $y(t) = [y_1(t), \dots, y_J(t)]^T$ obtenu par transformation de x par une fonction $g(x) = [g_1(x), \dots, g_J(x)]^T$. La méthode de résolution pour obtenir les slow features consiste à déterminer g telle que $y_j(t) := g_j(x(t))$ avec $\Delta_j(y_j) = < y'^2_j >$ minimale sous les contraintes suivantes

$$< y_j > = 0 \ (moyenne\ nulle)$$
$$< y_j^2 > = 1 \ (variance\ unitaire)$$
$$\forall j' < j, < y_j, y_{j'} > = 0 \ (décorrélation)$$

Où $< f >= \frac{1}{t_1 - t_0} \int_{t_0}^{t_1} f(t)dt$ est la moyenne temporelle de f

Ces descripteurs y_j , avec des variations minimales, sont alors caractéristiques du mouvement et peuvent être utilisés pour l'interprétation de scène. Ces slow features peuvent être extraits directement de cuboïdes de la vidéo pour la reconnaissance d'actions humaines [15]. Pour la classification de scènes dynamiques naturelles (tornade, éclairs, avalanches..), les slow features ont été extraits à partir de filtres de Gabor de différentes échelles et orientations appliqués à une séquence d'image [16].
Cette extraction de caractéristiques peut être associée donc à des descripteurs existants.

La méthode la plus courante pour exprimer localement le mouvement apparent est le calcul du flot optique. Il est obtenu en résolvant l'équation de conservation du mouvement apparent (ECMA) comme détaillé dans l'étude bibliographique précédente. Le flot est un champs vectoriel composé de mouvements verticaux et horizontaux.

$$F : \quad \Omega \subset \mathbb{R}^2 \quad \rightarrow \quad \mathbb{R}^2$$
$$(x, y) \quad \rightarrow (U(x, y), V(x, y))$$

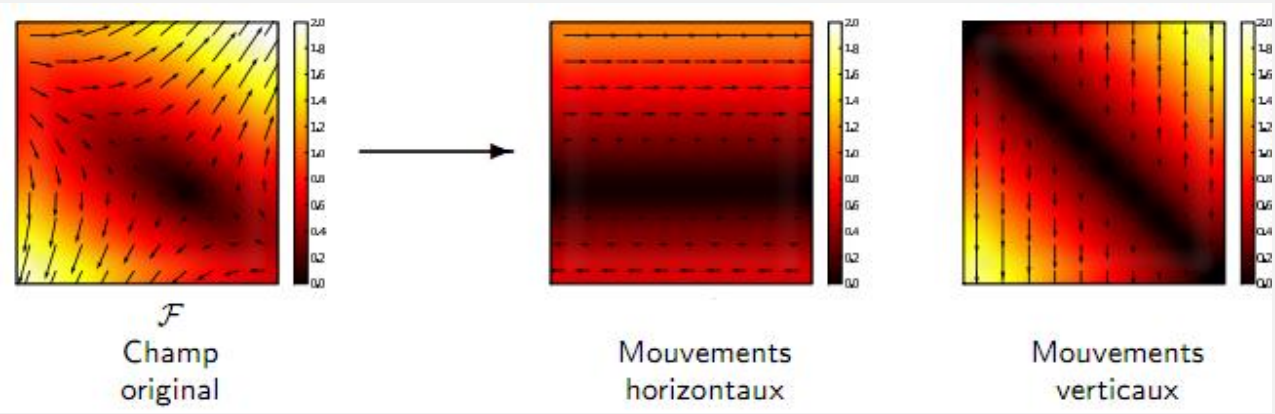

Illustration 14 : Décomposition d'un champ vertoriel en composantes verticales et horizontales

Le flot optique constitue une description de bas niveau de modélisation qui sert de base pour construire des descripteurs performants dans la reconnaissance de scène.

Kihl, Tremblais et Augereau [10] se sont inspirés de l'analyse des mouvements des fluides pour extraire des points singuliers représentatifs du mouvement local depuis un champ de déplacement. La chaîne d'extraction commence par la projection du champ de vecteurs sur l'espace des polynômes de degré 1

$$v = Ax + b$$

Avec v le vecteur approché du déplacement du point x.

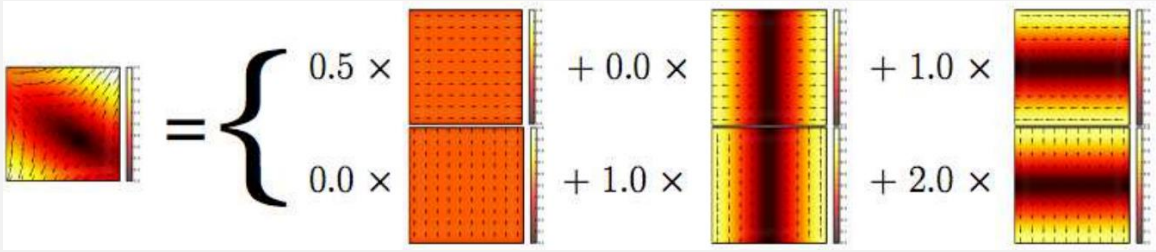

Illustration 15 : Exemple d'une décomposition d'une projection dans la base des polynômes de degré 1

Puis l'analyse des coefficients analytiques de la projection fournit la position d'un point critique $x = -A^{-1}b$ Le type de singularité de ce point critique est déterminé par les valeurs propres de A. Cette méthode, étant rapide et ne détectant au plus qu'une singularité, sera effectuée à différentes échelles de la vidéo pour détecter toutes les singularités présentes au temps t.

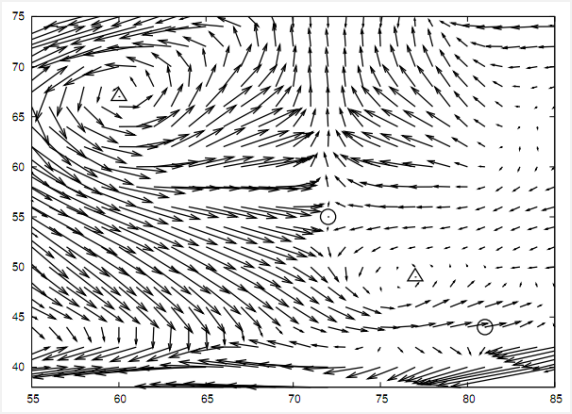

Illustration 16 : Exemple de singularités détectées dans un champ de déplacement

Cette méthode pourrait servir de détecteur pour ensuite effectuer une description autour de ces points regroupant l'information du mouvement.

Les tenseurs d'orientations construits à partir des coefficients de projections du flot optique sur l'espace des polynômes fournissent des descripteurs prometteurs pour la description du mouvement global dans une vidéo [11]. Un tenseur d'orientation est une représentation de l'orientation locale sous la forme d'une matrice symétrique de taille NxN pour un signal v de dimension N.

$$T = v.v^T$$

Combiné à un tenseur sur des descripteurs spatio-temporel HOG3D [12], le tenseur d'orientation sur le flot optique est particulièrement performant dans la reconnaissance d'actions sur la base KTH.

Dans cette partie, je vous présenterai une méthode d'approximation et de description des champs de déplacement pour la reconnaissance de mouvements simples [17] et son application pour la description de l'orientation des mouvements de tête d'un utilisateur devant un écran d'ordinateur.

2. Approximation d'un champ de vecteur par un polynôme

Un champ de déplacement à un temps t dans une séquence d'images est définit par :

$$\begin{aligned} F: \quad \Omega \subset \mathbb{R}^2 &\rightarrow \quad \mathbb{R}^2 \\ (x,y) &\rightarrow (U(x,y), V(x,y)) \end{aligned}$$

avec U et V les fonctions permettant d'obtenir le déplacement de point de coordonnées (x,y) ∈ Ω en horizontal et en vertical.

Notons E_p l'espace vectoriel des fonctions de Ω dans R contenant les fonctions U et V.

Munissons Ep du produit scalaire suivant :

$$< F_1, F_2 > = \iint_{\Omega} F_1 . F_2 . w(x,y) dx dy$$

avec w une fonction de poids choisie en fonction de notre problème. Nous avons choisi une fonction de poids constante de 1 et Ω =[-1,1] dans la suite de cette étude.

L'espace E_p muni de ce produit scalaire est alors un espace préhilbertien.

La norme associée à ce produit scalaire est alors :

$$||F|| = \sqrt{< F, F >}$$

De sorte à étudier ce champ de vecteurs, nous allons approximer chacune des composantes U et V par des combinaisons linéaires de fonctions polynomiales. Les polynômes bivariables de degré N sont des élèments de E_p de la forme:

$$P(x,y) = \sum_{k=0}^{N} \sum_{l=0}^{N-k} a_{kl}\, x^k y^l$$

Notons P_N l'ensemble des fonctions polynomiales bivariables de degré N. P_N est un sous-espace vectoriel de E_p de dimension finie $\frac{(N+1)(N+2)}{2}$. Considérons la base orthogonale des polynômes de Legendre définis par la formule de récurrence suivante :

$$\begin{aligned} L_{0,0} &= 1 \\ L_{1,0} &= x \\ L_{0,1} &= y \\ L_{i+1,j} &= \frac{2i+1}{i+1} x . L_{i,j} - \frac{i}{i+1} L_{i-1,j} \\ L_{i,j+1} &= \frac{2j+1}{j+1} y . L_{i,j} - \frac{j}{j+1} L_{i,j-1} \end{aligned}$$

A partir de cette base, nous définissons une base orthonormale B en normalisant chacun des polynômes pour la norme ||.|| .

Pour approcher une fonction F de E_p par une fonction de P_N, nous allons projeter orthogonalement F sur P_N. Puisque P_N est un sous-espace vectoriel de dimension finie de l'espace préhilbertien E_P et que $B = (e_i)_{1 \leq i \leq \frac{(N+1)(N+2)}{2}}$ est une base orthonormée de P_N, F possède une unique projection P_F orthogonale sur P_N dont l'expression dans la base B est

$$P_F = \sum_{i=0}^{D} \sum_{j=0}^{D-i} < F, L_{i,j} > L_{i,j}$$

Nous obtenons alors la décomposition de la projection dans la base des polynômes B.

3. Projection du flot optique

Le champ de déplacement obtenue à partir d'une vidéo est une fonction discrète définit par :

$$\begin{array}{rccc} C: & [0, r-1] \times [0, c-1] & \rightarrow & \mathbb{R}^2 \\ & (x, y) & \rightarrow & (U'(x,y), V'(x,y)) \end{array}$$

avec (r,c) la taille de la vidéo considérée.
Nous effectuons la transformation affine suivante pour obtenir un champ de vecteur de E_p:

$$\begin{array}{rccc} N: & [0, r-1] \times [0, c-1] & \rightarrow & [-1,1]^2 = \Omega \\ & (x_1, x_2) & \rightarrow & \begin{pmatrix} \frac{2}{r-1} & 0 \\ 0 & \frac{2}{c-1} \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} + \begin{pmatrix} -1 \\ -1 \end{pmatrix} = (x, y) \end{array}$$

Ainsi nous avons le nouveau champ de déplacement U=U'oN

Alors le produit scalaire $< U, e_i > = \int_{-1}^{1} \int_{-1}^{1} U(x,y) e_i(x,y) dxdy = \int_{\Omega} U' o N^{-1}(x,y) e_i(x,y) dxdy$

$$= \int_0^{r-1} \int_0^{c-1} U'(x_1, x_2). e_i o N(x_1, x_2). \frac{4}{(r-1)(c-1)}. dx_1 dx_2$$

Avec le changement de variables en utilisant la fonction N.

En discrétisant la formule nous obtenons alors :

$$< U, e_i > = \frac{4}{(r-1)(c-1)} \sum_{i=0}^{r-1} \sum_{j=0}^{c-1} U'(x_1, y_1). e_i o\, N(x_1, x_2)$$

Donc avec la construction de la base B, le calcul des produits scalaires et l'application du théorème de projection, nous obtenons la projection du flot optique sur l'espace des polynomes de degré N.

Cette projection possède deux principaux avantages :

- elle diminue considérable le taille de l'information caractéristique du mouvement : on passe de 2*r*c coefficients à $2 * \frac{(N+1)(N+2)}{2}$ coefficients
- le champ obtenu est un lissage du champ de départ ce qui permet donc de diminuer de façon implicite le bruit.

Par contre, du fait de ce lissage, un degré trop faible ne permet pas de modéliser des champs de vecteurs

complexes. Il faut donc adapter le degré des polynômes à l'application.

4. Analyse en composantes principales

Dans le but d'étudier les mouvements d'une séquence, nous allons analyser les variations dans le temps des coefficients dans la base B de la projection du champ vectoriel.
A partir d'une liste de coefficients extraits de plusieurs vidéos, nous avons effectué une analyse en composantes principales (ACP). Cette méthode permet de réduire des variables liées entre elles à un ensemble de variables décorrélées et ainsi de rendre l'information moins redondante.
On considère chaque composante du vecteur de coefficients dans une variable aléatoire $\left(X_1, \dots, X_{\frac{(N+1)(N+2)}{2}}\right)$ de moyenne nulle et de variance unitaire. Puis on utilise une base de vidéo de référence pour obtenir un échantillon représentatif des occurrences de ces variables stockés en ligne dans une matrice X. Pour obtenir une variance unitaire et une moyenne nulle, nous effectuons une transformation affine sur les échantillons fournit pour les normaliser. Puis on construit une base constituée des vecteurs propres de la matrice de covariance $X^T.X$ calculée grâce à cet échantillon. Dans ce nouveau système de coordonnées, un vecteur initial peut être approximer en prenant le début de la combinaison linéaire, constituée des vecteurs propres ayant les valeurs propres les plus élevées. En particulier cela consiste à effectuer une projection du vecteur initial sur le sous-espace vectoriel ayant pour base les vecteurs propres ayant les valeurs propres dominantes. Nous pouvons alors représenter notre vecteur de coefficients par un ou deux coefficients.

5. Application à la reconnaissance de l'orientation des mouvements de tête

Dans notre problématique de caractériser l'orientation des mouvements de la tête d'un utilisateur face à un écran d'ordinateur, nous utiliserons les polynômes de degré 2 et une ACP en 1 coefficient représentatif puisque les mouvements et les descriptions sont simples. Comme une vidéo de 2 secondes fournit déjà 60 frames et donc 59 échantillons de coefficients, nous avons construit notre propre base de mouvements de tête pour cette application.

Vous pouvez voir dans l'illustration suivante l'interface du programme constituée de gauche à droite de la sortie de la webcam en direct, le flot optique extrait et de sa projection polynomiale. La projection est très lisse comparée au flot initial comme vous pouvez le voir, car notre degré est de 2 donc assez bas. En-dessous se trouve l'évolution des coefficients u pour les mouvements horizontaux et v pour les verticaux. Lors de cette série, j'ai effectué successivement des mouvements vers la droite et vers la gauche dont quelques sont annotés dans l'illustration.

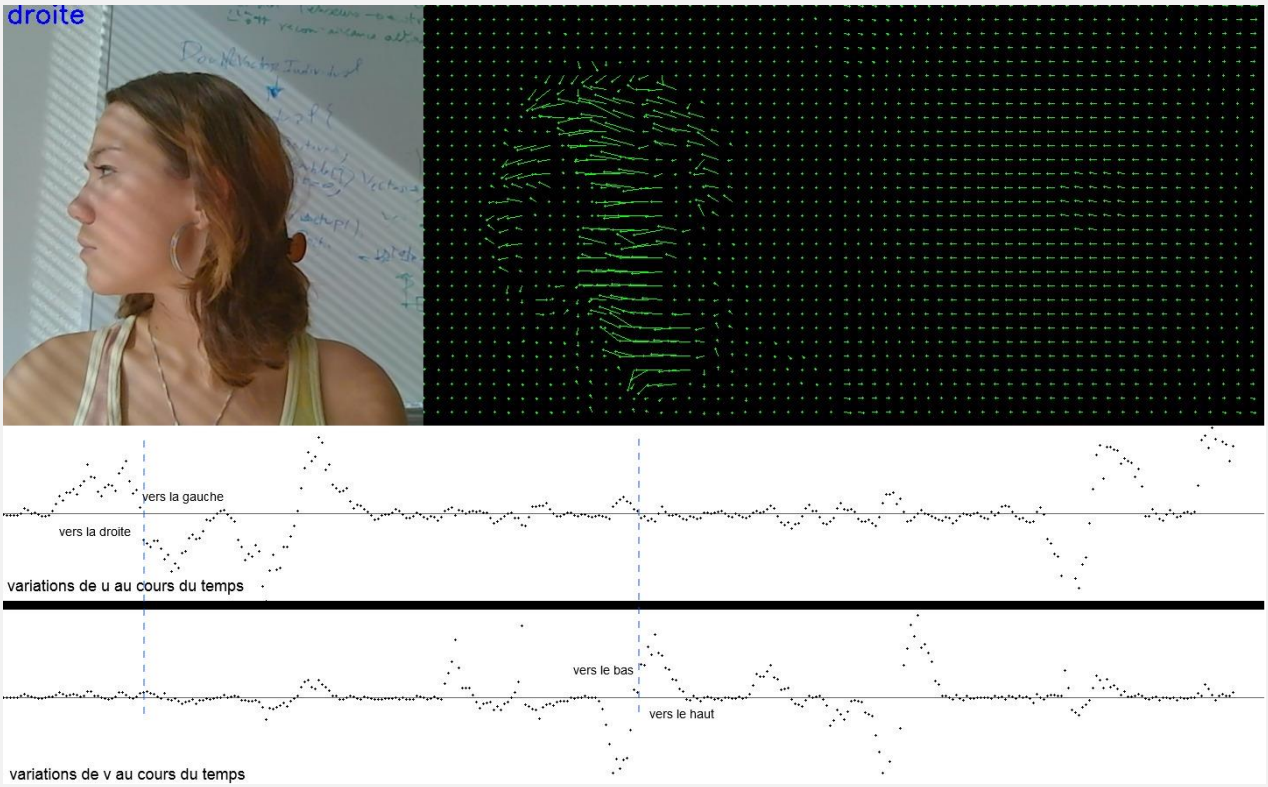

Illustration 17 : interface du programme de reconnaissance d'orientation de mouvements de tête

Pour reconnaître les quatre orientations principales et la position statique, nous avons construit un descripteur pour chaque couple de frame constituée de 6 coefficients : les deux coefficients obtenus après avoir effectué la projection par ACP au temps t, t-1 et t-2. Nous avons ensuite effectué une classification par un unique SVM pour ces 5 classes.
Nous avons choisi ce descripteur $(u_{t-2}, v_{t-1}, u_{t-1}, v_{t-1}, u_t, v_t)$ car nous avions plus de 10% de reconnaissance qu'avec simplement un descripteur de type (u_t, v_t).

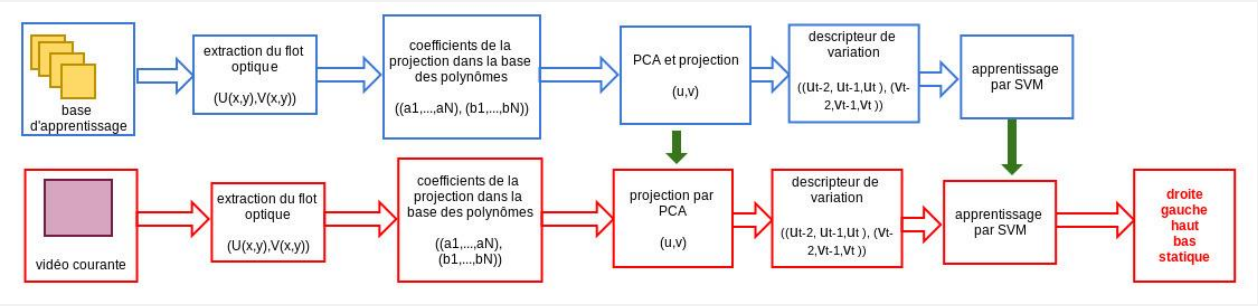

Illustration 18 : chaine de traitement de programme de reconnaissance

Conclusion

La participation au concours FishClef fut un réel challenge dans la construction d'un système de reconnaissance dans les vidéos de la base d'apprentissage étaient difficiles à traiter dû à la faible résolution, aux changements de luminosité sous-marins et aux mouvements permanents des algues liées aux courants. J'ai pu ainsi apprécier la recherche en vision par ordinateur en me renseignant sur les méthodes utilisées pour la description de contenu vidéo et les moyens à envisager pour traiter ce type de difficultés. Le programme a été conçu de façon à répondre au mieux aux critères de mesure de performance de la détection et de la reconnaissance imposés par les organisateurs du concours. L'algorithme devant gérer une lourde base de données vidéos, les tests de programmation et les visualisations ont été effectués localement tandis que les grandes étapes du système comme l'extraction des descripteurs, l'apprentissage et la construction des fichiers de soumission ont été effectués à distance sur des serveurs de calculs tout en contrôlant le temps de calcul du programme et en parallélisant les tâches dès que possible. De plus, certaines tâches comme la détection et la reconnaissance dans une vidéo prenaient plus de 24h alors j'ai dû analyser la distribution des tailles des vidéos de sorte à anticiper en estimant le temps nécessaire à la production des fichiers à soumettre. Enfin j'ai pu écrire plusieurs types d'articles : une note de travail, une description insérée dans un article écrit en collaboration et un article suivant le format demandé pour la conférence MAED ; et prendre en considération les remarques des relecteurs pour les améliorer.
Le système de reconnaissance pourrait être amélioré de nombreuses façons. Le tracking permettrait d'extraire plus de descripteurs des espèces dans la base d'apprentissage. Un premier SVM pourrait servir à différencier les descripteurs de poisson en général du background, puis les descripteurs de poissons seraient ensuite classifier par les différentes espèces. Rétrécir les boîtes englobantes permettrait d'augmenter le rappel moyen du système. Le descripteur Opponent Sift pourrait être associé à un descripteur de mouvement de sorte à différentier les poissons par leurs formes et leurs couleurs ainsi que par leurs déplacements. Enfin il est possible d'exploiter des metadata founies dans la base d'apprentissage comme les données GPS de la capture de la vidéo par exemple.

L'approximation du flot optique par projection sur l'espace des polynômes m'a permis d'analyser des articles et de les appliquer précisément dans le but d'obtenir des résultats semblables en appliquant des discrétisations et en vérifiant le fondement mathématique de ces méthodes. Cela m'a permis d'appréhender l'acquisition et le traitement effectué sur le flot optique que j'utiliserai par la suite. À la place de l'ACP, il serait intéressant de tester la représentation par SFA des coefficients des projections ce qui fournirait un signal non seulement décorrélé mais également plus stable au cours du temps, et donc permettrait de raccourcir le descripteur d'apprentissage. De plus, nous obtenons les variations des coefficients u et v au cours du temps et donc les mouvements effectués dans le temps et non la position de la tête par rapport à la position de référence. Pour étudier la position de la tête de l'utilisateur, il faudrait calculer les intégrales des variations de coefficients.

Bibliographie

1. **On Space Time Interest Points**, Yvan Laptev, International Journal of Computer Vision, 2005
2. **Evaluation of local spatio-temporal features for action recognition,** Heng Wang, Muhammad Muneeb Ullah, Alexander Kläser, Ivan Laptev et Cordelia Schmid, Proceedings of the British Machine Vision Conference, 2009
3. **CAMSHIFT improvement on multi-hue object and multi-object tracking,** Hidayatullah P. and Konik H., *Visual Information Processing (EUVIP), 2011 3rd European Workshop on*
4. **Two-frame motion estimation based on polynomial expansion**, Gunnar Farneback , Lecture Notes in Computer Science, 2003
5. **Object recognition from local scale-invariant features**, Lowe, D.G., *Computer Vision, 1999. The Proceedings of the Seventh IEEE International Conference on*
6. **Evaluating Color Descriptors for Object and Scene Recognition**, Koen E. A. van de Sande, Theo Gevers and Cees G. M. Snoek, *IEEE Transactions on Pattern Analysis and Machine Intelligence*
7. **Improved adaptive Gausian mixture model for background subtraction,** Z.Zivkovic, International Conference Pattern Recognition, UK, August, 2004
8. **ViBe: A Universal Background Subtraction Algorithm for Video Sequences,** Barnich, O.; Van Droogenbroeck, M., Image Processing, IEEE Transactions on,Vedaldi, A., Fulkerson, B.:
9. **VLFeat - an open and portable library of computer vision algorithms**. In: ACM International Conference on Multimedia. (2010)
10. **Multivariate orthogonal polynomials to extract singular points**, Kihl, O.; Tremblais, B.; Augereau, B., *Image Processing, 2008. ICIP 2008. 15th IEEE International Conference on* ,
11. **A Tensor Motion Descriptor Based on Multiple Gradient Estimators,** Sad, D.; Mota, V.F.; Maciel, L.M.; Vieira, M.B.; De Araujo, AA, *Graphics, Patterns and Images (SIBGRAPI), 2013 26th SIBGRAPI - Conference on*
12. **A tensor based on optical flow for global description of motion in videos,** V.F. Mota, E.A. Perez, M.B. Vieira, L.M. Maciel, F. Precioso et P.H. Gosselin, IEEE SIBGRAPI Conference on Graphics, Patterns and Images, 2012
13. **Slow Feature Analysis: Unsupervised Learning of Invariances**, Wiskott, L; Sejnowski, T, *Neural Computation*, 2002
14. **Slow Feature Analysis for Human Action Recognition**, Zhang Zhang; Dacheng Tao, *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 2012
15. **Dynamic Scene Classification: Learning Motion Descriptors with Slow Features Analysis**, Theriault, C.; Thome, N.; Cord, M., *Computer Vision and Pattern Recognition (CVPR), 2013 IEEE Conference on*
16. **Modélisation de champs de vecteur par bases de polynômes : application à l'analyse de la posture d'utilisateurs devant un écran d'ordinateur, via une webcam,** M. Druon, B. Tremblais, B. Augereau

ANNEXE

Description de l'entreprise

Le Laboratoire i3s est l'un des plus gros laboratoires de la Côte d' Azur, situé plus précisément dans la plus grande technopole de France, Sophia-Antipolis. Les recherches qui y sont effectuées se trouvent dans le domaine des sciences de l'information et de la communication en partenariat avec le CNRS et l'Université de Nice Sophia Antipolis.
Le laboratoire est organisé en 4 pôles scientifiques dont le pôle GLC (Génie du Logiciel et de la Connaissance) dans lequel j'effectue ce stage. Ce pôle recouvre en particuliers les aspects liés :

- à la maîtrise de la complexité logicielle
- la dynamicité et l'adaptabilité, en particulier en fonction de l'évolution du contexte d'exécution
- la globalisation des calculs et leur déploiement sur des infrastructures distribuées
- la description sémantique des processus et des données, la construction et l'utilisation des bases de connaissances.

Mon stage se concentre surtout sur ce dernier aspect.

Le pôle GLC est constitué de quatre équipes dont l'équipe MinD que j'ai intégré en tant que stagiaire. Les activités de recherches de l'équipe MinD concerne principalement les algorithmes pour l'exploration de données et l'apprentissage appliqués à plusieurs domaines comme le multimédia, la biologie, le texte...

Description des maîtres de stage

Diane Lingrand, enseignante-chercheuse qui s'intéresse aux algorithmes de traitement d'images et de vision associées aux méthodes d'apprentissage. Elle s'intéresse notamment à la reconnaissance de plantes et des poissons et, dans le cadre d'une collaboration avec l'entreprise Capabilis, à la reconnaissance de sécurités visuelles dans les documents administratifs.

Frédéric Precioso, professeur des universités qui s'intéresse aux méthodes de SVM (support vector machine) avec noyaux, au boosting, aux méthodes d'apprentissage sur de grosses bases de données et aux méthodes pour la reconnaissance de formes. Il utilise ces méthodes notamment dans le contexte de la classification multimédia (images, vidéos, …), de la classification des sens dans le texte et dans des applications pour la vision par ordinateur.

Résumé

Durant ce stage découpé en deux missions principales, nous nous sommes intéressés à la description et à l'interprétation de contenu vidéo.
Tout d'abord, nous avons construit une chaîne de traitement capable de détecter et de reconnaitre des poissons dans une séquence d'image récupérée de vidéo-surveillance sous-marines lors de notre participation au concours FishClef. Notre système de reconnaissance se base sur la modélisation de l'arrière-plan d'une vidéo par une somme pondérée de lois gaussiennes, l'extraction dense de points clés à large échelle et leur description par Opponent Sift et enfin l'apprentissage de ces données par des Support Vector Machine.
Puis nous nous sommes concentrés sur l'extraction de caractéristiques du mouvement global d'une vidéo à partir de son flot optique. Pour cela, nous avons projeté le flot optique sur l'espace vectoriel des polynômes, puis effectuer une analyse en composante principale pour diminuer la taille de la représentation. Ce descripteur a été testé dans l'application à la reconnaissance d'orientation de mouvement de tête dans le but d'effectuer ensuite une reconnaissance de la posture d'un utilisateur face à un ordinateur.

Abstract

During this internship divided in two main missions, we were interested in the description and in the interpretation of video contents.
First of all, we built a processing chain capable of detecting and of recognizing fishes in a marine video from underwater video monitoring during our participation in the FishClef challenge. Our recognition system is based on the modelling of the background of a video by a Gaussian mixture model, the dense extraction of large scale keypoints and their description by Opponent Sift and finally the learning of these data by Support Vector Machine.
Then we concentrated on the extraction of characteristics of the global movement of a video from its optical flow. For this purpose, we projected the optical flow on the vectorial space of polynomials, then perform a principal component analysis in order to reduce the representation dimension. This descriptor was applied to the recognition of orientation of head motion in order to later recognize the head pose of a user in front of a computer.