

Détection par mesures de similarité discriminatives

4.1 Introduction

Les systèmes de détection d'objets et en particulier les systèmes de détection de visages de l'état de l'art sont tous basés sur des mesures de similarité discriminatives. Malgré la difficulté induite par la nécessité de représenter la classe 'non objet', ces méthodes semblent permettre une meilleure classification que les méthodes génératives. La corrélation, associée à la méthode des plus proches voisins, a permis d'obtenir un système de détection fonctionnel mais dont les résultats semblent limités par la méthode de classification choisie. Afin d'améliorer les taux de détection tout en utilisant toujours une base d'exemples de dimension réduite, nous proposons, dans ce chapitre, d'utiliser les méthodes mises au point pour les systèmes de détection par corrélation en les adaptant à une mesure de similarité discriminative plus performante. Pour ce faire, nous avons remplacé la corrélation par un Perceptron Multicouche. Nous avons ensuite appliqué les deux idées qui ont donné les meilleures améliorations à la méthode par corrélation, c'est à dire, l'association de classifieurs et l'utilisation de filtres détecteurs de contours. Nous montrerons à la fois l'efficacité de l'utilisation d'un Perceptron Multicouche en lieu et place d'une simple corrélation, mais aussi que nos conclusions sur les méthodes que nous utilisons avec la corrélation se généralisent à des classifieurs plus complexes et performants. Nous commencerons par décrire un système de détection basé sur un MLP et montrerons l'influence de la forme du MLP choisie, du nombre d'exemples et de la méthode de BootStrapping. Nous appliquerons ensuite les méthodes que nous avons mises au point précédemment et montrerons comment elles influencent les résultats sur le problème particulier de la détection de visages.

4.2 Système de Détection basée sur un MLP avec peu d'exemples

Dans cette section nous utilisons un Perceptron Multicouche appliqué aux images en Niveaux de Gris, au problème de la détection de visages. Nous commencerons par décrire le fonctionnement d'un tel système, puis nous donnerons différents résultats expérimentaux sur la base de test de détection de visages CMU. Nous nous intéresserons à l'influence du nombre d'exemples, à la forme du MLP ainsi qu'à la normalisation des images.

4.2.1 Fonctionnement du système de détection

Le système de détection que nous employons ici est un système de détection 'classique' (figure : 1.2) où nous utilisons un MLP directement appliqué aux images en Niveaux de Gris comme classifieur. Nous nous servons d'une rétine de 25×25 pixels afin que notre système puisse fonctionner sur les bases de test standards de détection de visages. La pyramide d'images permettant la détection multi-échelle utilise un facteur de sous-échantillonnage de 1.2.

Nous commencerons par utiliser un MLP à trois couches entièrement connecté (figure : 2.15) avec n_2 neurones dans la couche cachée. Le nombre n_1 de neurones de la première couche est directement lié à la dimension de la rétine puisque l'entrée du réseau est un vecteur représentatif de l'image contenue dans la rétine. La troisième et dernière couche comporte un seul et unique neurone. Le réseau de neurones est entraîné de façon à retourner la valeur 1 si l'image d'entrée représentée par le vecteur \mathbf{x} est un visage (ou plus généralement l'objet à détecter) et -1 dans le cas contraire. Les fonctions d'activation utilisées sont les fonctions tangentes hyperboliques (figure : 2.16).

4.2.1.1 Entraînement du perceptron

Les N premiers exemples de la base de visages 'Caltech WebFaces' sont utilisés comme images de référence. Une base de 450 images ne contenant pas de visage est utilisée pour l'algorithme de BootStrapping. Afin de minimiser les risques de sur-apprentissage liés au faible nombre d'exemples ainsi que d'obtenir un système d'apprentissage plus rapide qu'avec l'utilisation classique de la méthode de BootStrapping, nous avons mis au point une variante de cette méthode.

Comme pour l'algorithme de BootStrapping classique, nous utilisons les fausses détections obtenues sur les images ne contenant aucun visage pour générer la classe 'non visage'. Cependant les conditions d'arrêt de l'algorithme, ainsi que la méthode d'apprentissage utilisée diffèrent de la méthode originale. L'algorithme classique de BootStrapping consiste à obtenir un certain nombre d'exemples de 'non visage' puis d'utiliser un algorithme d'apprentissage, généralement l'algorithme de Backpropagation pour les MLP. Cette opération étant répétée un certain nombre de fois, généralement jusqu'à ce que l'ajout de nouveaux exemples n'améliore plus le taux de détection du système.

Ici, l'apprentissage et l'algorithme de BootStrapping sont plus imbriqués que dans l'algorithme classique. L'algorithme de Backpropagation Stochastique utilisé ne diverge de l'algorithme classique (Algo : 5) que par les conditions d'arrêt et la sélection des exemples de 'visage' et de 'non visage'.

La condition d'arrêt consiste à stopper l'apprentissage lorsque l'on atteint un nombre donné d'itérations max_{iter} sans vérifier la convergence de l'algorithme de Backpropagation. La valeur de max_{iter} est relativement faible, généralement égale au nombre d'exemples utilisés pour l'apprentissage.

De façon à donner autant d'importance à la classe 'visage' qu'à la classe 'non visage' dont les dimensions peuvent être très différentes, l'algorithme alterne systématiquement entre un exemple de 'visage' aléatoirement choisi et un exemple de 'non visage'.

L'algorithme de Backpropagation stochastique est alors défini comme suit :

Algorithm 8 Stochastic Backpropagation V2

```

1: initialisation :  $\eta, \mathbf{w}, n_2, \theta, max_{iter}, m = 0$ 
2: while  $m < max_{iter}$  do
3:    $m = m + 1$ 
4:   if  $m$  pair then
5:      $\mathbf{x} \leftarrow$  échantillon 'visage' exemple choisi aléatoirement
6:   else
7:      $\mathbf{x} \leftarrow$  échantillon 'non visage' exemple choisi aléatoirement
8:   end if
9:    $\delta_k = (t_k - z_k) \varphi'_k(net_k)$ 
10:   $\delta_j = [\sum_{k=1}^c \delta_k w_{kj}] \varphi'_j(net_j)$ 
11:   $w_{kj} = w_{kj} + \eta \delta_k y_j$ 
12:   $w_{ji} = w_{ji} + \eta \delta_j x_i$ 
13: end while
14: return  $\mathbf{w}$ 

```

Ainsi, une seule utilisation de l'algorithme de Backpropagation est insuffisante pour converger vers une solution de classification optimale au sens des moindres carrés.

Le but de notre algorithme de BootStrapping que nous nommerons Online BootStrapping est de minimiser le taux de fausses détections (FAR) sur les images de la base ne contenant aucun visage. Pour ce faire, chaque erreur détectée par le classifieur est ajoutée à la base de 'non visage'. Lorsque le nombre de fausses détections atteint un nombre N_{fdmax} donné (nous avons choisi $N_{fdmax} = 100$), nous entraînons le réseau de neurones à l'aide de l'algorithme de Backpropagation (Algo : 8). Afin de se focaliser sur les exemples de 'non visage' les plus significatifs, nous limitons le nombre total N_{ft} d'exemples de 'non visage' à $N_{ftmax} = 3000$. Si le nombre d'exemples de 'non visage' est supérieur à N_{ftmax} , alors nous supprimons les échantillons les moins pertinents, *i.e.*, ceux dont le score donné par le classifieur s'approche le plus de la valeur souhaitée (-1). L'algorithme de BootStrapping s'arrête lorsque le taux de fausses détections (ou fausses alarmes) FAR est inférieur à un seuil FAR_{min} donné.

Le taux de fausses détections est calculé tous les $N_{test} = 2 \times 10^6$ tests et correspond au rapport du nombre de fausses détections effectuées N_{fd} sur le nombre de tests, *i.e.*, $FAR = \frac{N_{fd}}{N_{test}}$.

Le fonctionnement de l'algorithme d'Online BootStrapping se résume donc ainsi :
 N_{ftmax} : nombre maximum d'échantillons de 'non visage'.
 N_{ft} : nombre total d'échantillons de 'non visage' utilisés pour l'apprentissage.

1. Réunir un petit nombre d'échantillons exemples de la classe 'non visage' et un ensemble d'échantillons représentatif de la classe 'visage'
2. Entraîner le MLP à partir de la base d'échantillons exemples courante par l'algorithme de Backpropagation.
3. Supprimer les échantillons de 'non visage' dont le score de classification est le plus proche de '-1' de façon à ce que $N_{ft} \leq N_{ftmax}$.
4. Utiliser le système de détection sur une base d'images ne contenant pas la classe à détecter. Obtenir N_{fdmax} fausses détections supplémentaires.
5. Retourner à l'étape 2 jusqu'à ce que $FAR = \frac{N_{fd}}{N_{test}} < FAR_{min}$

Cette méthode permet ainsi, non seulement une convergence rapide du classifieur en le focalisant au maximum sur les exemples les plus significatifs mais permet en plus, grâce à la condition d'arrêt utilisée de minimiser les risques de sur-apprentissage, ce qui est nécessaire à notre système qui doit être capable de fonctionner avec peu d'exemples.

Afin de mesurer l'efficacité de la méthode d'apprentissage, nous observerons trois paramètres qui nous permettront de caractériser les performances de cet algorithme.

1. Le Rappel (R) : le nombre de visages exemples dont le MLP renvoie un score supérieur à zéro sur le nombre total d'exemples.
2. Le taux de fausses alarmes (FAR) : $FAR = \frac{N_{fd}}{N_{test}}$. $N_{test} = 2 \times 10^6$
3. Le nombre de fichiers images ne contenant pas de visages utilisés pour converger vers une solution ($NbIm$) (cela nous permet de caractériser la vitesse de convergence de la méthode)

4.2.1.2 Regroupement des résultats

Afin d'éviter les détections superposées, nous utiliserons une méthode qui a montré son efficacité, notamment dans le cadre du détecteur de visages de Garcia et Delakis.

Chaque rétine 25×25 dans la pyramide d'images pour laquelle le classifieur renvoie un score supérieur à zéro est considérée comme une détection. Le système de regroupement choisit la détection avec le score le plus proche de 1. L'ensemble des détections superposées à cette dernière est alors regroupé en une seule détection dont l'échelle E et la position \mathbf{p} sont les moyennes pondérées des détections superposés. Le score s_r , correspond à la somme des scores de détection. Ainsi, plus le nombre de

superpositions avec un score proche de 1 est important, plus le score de détection final sera élevé. L'opération est ensuite réitérée jusqu'à ne plus avoir de superpositions.

Si nous avons K détections superposées, E_i , \mathbf{p}_i et s_i représentant respectivement l'échelle, la position et le score de la i^{ieme} détection superposée, alors :

$$s_r = \sum_{i=0}^{i<K} s_i \quad (4.1)$$

$$E = \frac{1}{s_r} \sum_{i=0}^{i<K} s_i E_i \quad (4.2)$$

$$\mathbf{p} = \frac{1}{s_r} \sum_{i=0}^{i<K} s_i \mathbf{p}_i \quad (4.3)$$

4.2.2 Résultats expérimentaux

Comme pour la corrélation, de nombreux paramètres peuvent influencer sur les résultats d'un système de détection basé sur un MLP. Nous pouvons citer en tout premier lieu, la forme du réseau de neurones, les traitements appliqués à l'image en Niveaux de Gris ou encore le nombre d'exemples de la base d'apprentissage. Nous proposons dans cette section d'observer les influences de ces différents paramètres sur notre système de détection. Nous commencerons comme pour la corrélation, par observer l'influence de la normalisation des vecteurs représentant les images en Niveaux de Gris, puis nous observerons les effets de la variation du nombre de neurones cachés du MLP, du nombre d'exemples d'apprentissage et finalement de la forme du MLP.

Afin d'effectuer ces tests nous utilisons la base 'Caltech WebFaces' comme base d'images de visages exemples. Plus exactement, les N échantillons exemples utilisés pour l'apprentissage correspondent aux N premiers visages extraits de cette base.

4.2.2.1 Influence de la normalisation

Nous avons vu, pour la corrélation que l'utilisation directe des images en Niveaux de Gris, sans normalisation, était impossible avec une mesure de similarité aussi simple. Dans cette section, nous proposons d'observer l'influence de la normalisation sur les taux de détection d'un système utilisant un réseau de neurones comme mesure de similarité. Pour ce faire, nous effectuerons les tests de détection de visages, sur la base CMU.

D'une certaine manière, un MLP est un système de détection par corrélations. En effet, ce dernier classe les images d'entrée grâce à diverses corrélations effectuées par chaque neurone du réseau. La normalisation $L2$ centrée donnant de meilleurs résultats que la normalisation simple pour la corrélation, c'est donc la première que nous utiliserons sur les vecteurs d'entrée du MLP.

Ainsi, si $\mathbf{x} = (x_1 \dots x_d)^T$ est le vecteur de dimension d qui représente l'image en Niveaux de Gris, le vecteur d'entrée normalisé \tilde{x} s'écrit :

$$m = \frac{1}{d} \sum_{i=1}^d x_i \quad (4.4)$$

$$\sigma = \sqrt{\sum_{i=1}^d (x_i - m)^2} \quad (4.5)$$

$$\tilde{x} = \frac{\mathbf{x} - m}{\sigma} \quad (4.6)$$

$$\mathbf{x} - m = (x_1 - m, \dots, x_d - m)^T \quad (4.7)$$

Les paramètres du système de détection n'étant pas indépendants les uns des autres, nous proposons d'effectuer les tests avec peu d'exemples, *i.e.*, 80, puis 1000 exemples ainsi qu'avec 10 puis 40 neurones cachés. De cette manière, nous obtiendrons des résultats pour des valeurs extrêmes des paramètres du système et en tirerons des conclusions assez générales sur l'influence de la normalisation.

Les courbes Rappel Précision obtenues (figure : 4.1) nous montrent que comme pour la corrélation, la normalisation est indispensable au bon fonctionnement du système de détection. En effet, sans la normalisation des images en Niveaux de Gris, les résultats de notre système de détection sont très nettement inférieurs à ceux obtenus avec la normalisation. Nous pouvons aussi constater que l'utilisation d'un plus grand nombre d'exemples d'apprentissage améliore les résultats, avec ou sans l'utilisation de la normalisation. Cependant, l'utilisation de plus de neurones cachés ne garantit pas une amélioration des résultats, en particulier, avec 1000 exemples, les résultats sans normalisation sont supérieurs avec 10 neurones cachés qu'avec 40.

Le tableau 4.1 résume les caractéristiques concernant la convergence du MLP en fonction des paramètres du système de détection (nombre de neurones cachés n_2 , nombre d'exemples N et normalisation). Sans normalisation, nous constatons que pour 80 exemples, le Rappel est très bon puisque supérieur à 90% et même égal à 1 dans le cas où nous utilisons dix neurones cachés. Cependant, étant donné le faible nombre d'exemples d'apprentissage, ce taux n'est pas très significatif et nous pouvons constater que les résultats du système de détection sont assez modestes. Lorsque nous utilisons mille exemples d'apprentissage, le Rappel n'est plus que de l'ordre de 70% alors qu'il est de plus de 95% avec la normalisation, ce qui explique la différence entre les résultats du système de détection avec et sans la normalisation.

En ce qui concerne le temps nécessaire à la convergence de l'algorithme d'Online BootStrapping, nous pouvons remarquer une tendance générale à l'augmentation du nombre de fichiers images (*NbIm*) nécessaires avec l'augmentation du nombre d'exemples d'apprentissage et avec le nombre de neurones cachés. Cependant, la valeur de (*NbIm*) reste assez variable et aléatoire.

4.2.2.2 Influence du nombre de neurones cachés

Nous avons vu, dans la section précédente que la normalisation est indispensable pour que l'algorithme de Backpropagation converge vers une solution qui per-

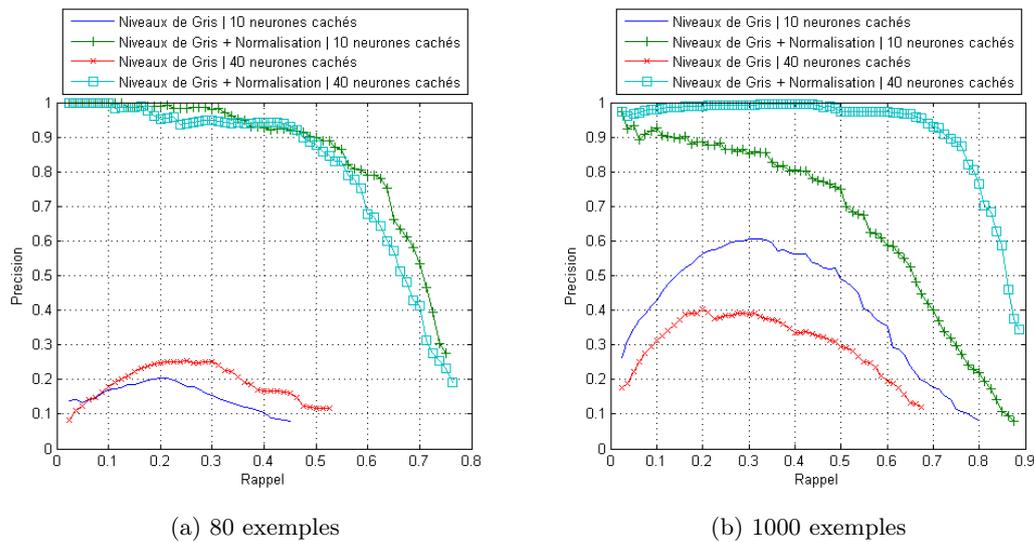


FIGURE 4.1 – Courbes Rappel Précision du système de détection basé sur un MLP et les images en Niveaux de Gris. Influence de la Normalisation. La normalisation apporte un gain substantiel aux résultats du système de détection.

Neurones Cachés	Niveaux de Gris				Normalisation			
	10		40		10		40	
Visages Exemples	80	1000	80	1000	80	1000	80	1000
Rappel (%)	100	71, 2	92, 5	74, 3	99, 4	99, 4	93, 5	97, 5
FAR ($\times 10^{-6}$)	46	47	21	46	48	42	46	40
NbIm	137	456	414	1126	26	460	110	330

TABLE 4.1 – Performances de l’algorithme d’apprentissage du MLP en fonction de la normalisation des vecteurs d’entrée, du nombre de neurones cachés et du nombre d’exemples de la base de référence.

met d’obtenir un système de détection fonctionnel. Avec 80 exemples, nous obtenons les mêmes résultats en utilisant 10 ou 40 neurones cachés. Cependant, nous voyons qu’avec 1000 exemples, l’augmentation du nombre de neurones cachés apporte un gain substantiel au niveau des taux de détection. Dans cette section, nous souhaitons mettre en évidence l’influence du nombre n_2 de neurones cachés. Pour ce faire, nous utiliserons 1000 puis 4000 exemples dans la base d’apprentissage afin de ne pas limiter les taux de détection par la faiblesse du nombre d’exemples. Nous donnons figure 4.2 et 4.3 les courbes Rappel Précision ainsi que le Rappel du système de détection pour 31 fausses détections pour un nombre de neurones cachés compris entre 5 et 40. Pour n_2 compris entre 5 et 15, l’augmentation du nombre de neurones cachés entraîne globalement une amélioration des résultats. Nous pouvons cependant constater que lorsque nous utilisons 4000 Exemples de référence, les résultats du système de

détection sont meilleurs avec 5 neurones cachés qu'avec 10. Au delà de 15 neurones cachés, les résultats restent sensiblement les mêmes, démontrant ainsi l'efficacité de l'algorithme d'Online BootStrapping pour éviter le sur-apprentissage.

Ainsi, si nous utilisons suffisamment de neurones cachés, les résultats sont alors limités par d'autres facteurs comme les traitements appliqués à l'image en Niveaux de Gris, et surtout le nombre d'exemples de la base de référence. A l'inverse, si le MLP est sous-dimensionné, le système d'apprentissage ne peut converger vers une solution qui permet d'obtenir à la fois un Rappel et une Précision élevés.

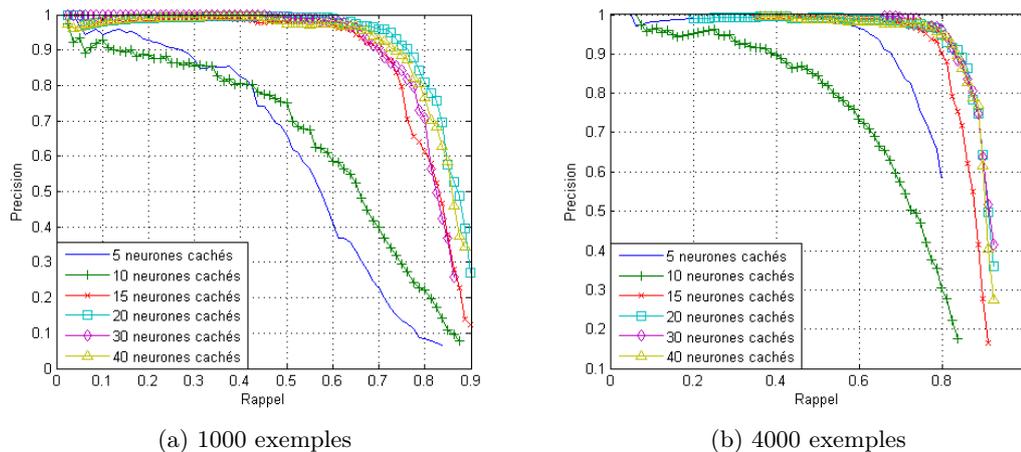


FIGURE 4.2 – Courbes Rappel Précision du système de détection basé sur un MLP en fonction du nombre de neurones cachés. Que ce soit pour 1000 ou 4000 exemples nous constatons une amélioration puis une stagnation des taux de détection avec l'augmentation du nombre n_2 de neurones cachés.

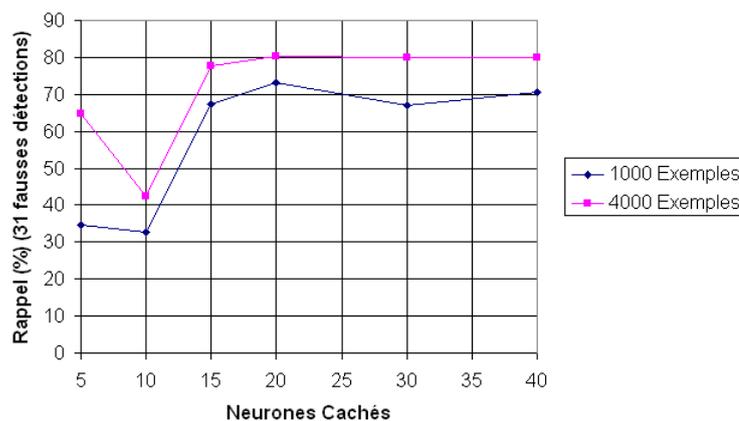


FIGURE 4.3 – Rappel du système de détection basé sur un MLP pour 31 fausses détections sur la base CMU en fonction du nombre de neurones cachés.

4.2.2.3 Influence du nombre d'images exemples

Nous avons pu constater grâce aux expérimentations précédentes que si nous dimensionnons convenablement le MLP, alors c'est le nombre d'exemples d'apprentissage qui limite la qualité de notre système de détection. Afin de mesurer les effets du nombre d'exemples d'apprentissage, nous avons appliqué le système de détection basé sur un MLP à 40 neurones cachés, à la base de test CMU, en faisant varier le nombre d'exemples d'apprentissage entre 80 et 5000. Les courbes Rappel Précision correspondantes sont visibles sur la figure 4.4. Nous pouvons constater une amélioration continue des résultats avec l'augmentation du nombre d'exemples. Afin de montrer la progression des résultats, nous avons reporté sur la figure 4.5 le Rappel du système de détection pour 31 fausses détections en fonction de la dimension de la base d'exemples de référence. Si 80 exemples ne permettent d'atteindre qu'un Rappel de 47% pour 31 fausses détections, ce qui est comparable aux résultats obtenus par le système de détection basé sur la corrélation, l'augmentation du nombre d'exemples améliore très rapidement les résultats, puisque nous atteignons un Rappel de 64% pour 300 exemples et de plus de 80% lorsque nous dépassons les 3000 exemples. Ainsi, malgré le fait que nous n'utilisons aucun traitement d'image complexe, nous constatons que notre système obtient des résultats qui se rapprochent de l'état de l'art lorsque nous utilisons un grand nombre d'exemples d'apprentissage. En effet, le système de Rowley *et al* obtient un Rappel de 86% et celui de Viola-Jones 85% pour 31 fausses détections sur la base CMU, contre un Rappel de 82% pour notre système de détection. La plupart des systèmes de l'état de l'art utilisent environ 4000 exemples d'apprentissage afin d'entraîner un classifieur discriminatif, ces résultats nous montrent que les performances des systèmes de détection sont principalement dues au nombre d'exemples et au type de classifieur utilisé.

4.2.2.4 Influence de la forme du MLP

Nous avons jusqu'à présent utilisé un MLP à trois couches entièrement connecté. Rowley *et al* dans [42] obtiennent des résultats au niveau de l'état de l'art en détection de visages avec un MLP à trois couches partiellement connecté. La forme du réseau de neurones est décrite section 2.3.7.1 et figure 4.6. Cette forme de réseau de neurones permet de réduire le nombre de connections et donc de diminuer à la fois les temps d'apprentissage et de détection. Ainsi, là où un MLP entièrement connecté à 26 neurones cachés compte 16276 connections, le MLP partiellement connecté de Rowley en compte 2078 pour le même nombre de neurones cachés et une rétine de 25×25 pixels, c'est à dire, près de huit fois moins de connections. Dans cette section, nous utilisons la forme de MLP de Rowley *et al* et comparons les résultats à ceux obtenus avec un MLP entièrement connecté.

Dans [42] les neurones cachés sont copiés deux, voire trois fois afin d'augmenter le nombre de paramètres du réseau de neurones. Nous obtenons alors trois réseaux de neurones distincts avec 26, 52 et 78 neurones cachés que nous nommerons respectivement Rowley1, Rowley2 et Rowley3. En premier lieu, nous avons comparé les résultats du système de détection utilisant le réseau de neurones de Rowley avec

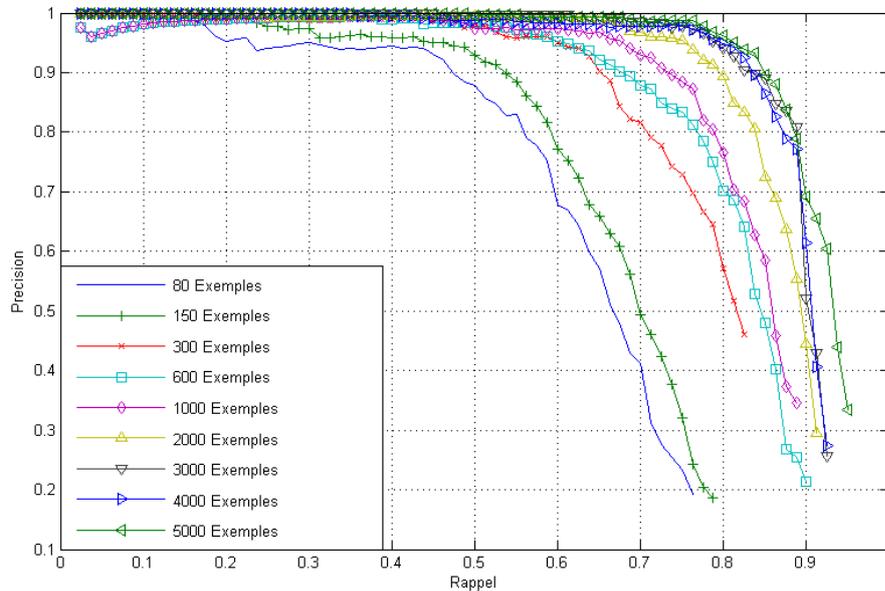


FIGURE 4.4 – Courbes Rappel Précision du système de détection basé sur un MLP entièrement connecté à 40 neurones cachés en fonction du nombre d'exemples de la base de référence. Plus nous utilisons d'exemples, meilleurs sont les taux de détection.

150, 1000 et 5000 exemples d'apprentissage et ceci pour 26, 52 et 78 neurones cachés. Nous pouvons constater (figure : 4.7) que les meilleurs résultats sont obtenus par les réseaux Rowley2 et Rowley3 qui comptent un total de $2078 \times 2 = 4156$ et $2078 \times 3 = 6234$ connexions. Si l'utilisation de 78 neurones cachés semble donner de meilleurs résultats, l'utilisation du réseau Rowley2 semble un bon compromis entre la complexité du réseau de neurones et les résultats du système de détection. En effet, nous obtenons avec un MLP possédant 4156 connexions des résultats équivalents à un MLP à 40 neurones cachés comptant 25040 connexions. Ainsi, nous en déduisons que c'est l'information locale qui est la plus pertinente pour distinguer un visage du 'reste du monde'.

En ce qui concerne le nombre de fichiers nécessaire à la convergence de l'algorithme (*NbIm*), il oscille entre 100 et 200 et est donc au moins deux fois inférieur au nombre nécessaire pour un MLP entièrement connecté. Cependant, comme pour un MLP entièrement connecté, *NbIm* varie de façon assez aléatoire et il est difficile de conclure sur l'influence du nombre d'exemples ou du nombre de connexions.

Dans la suite des expérimentations, nous utiliserons le MLP partiellement connecté Rowley2, qui est un bon compromis entre les résultats et les temps de calculs nécessaires à l'apprentissage et à l'évaluation du système de détection.

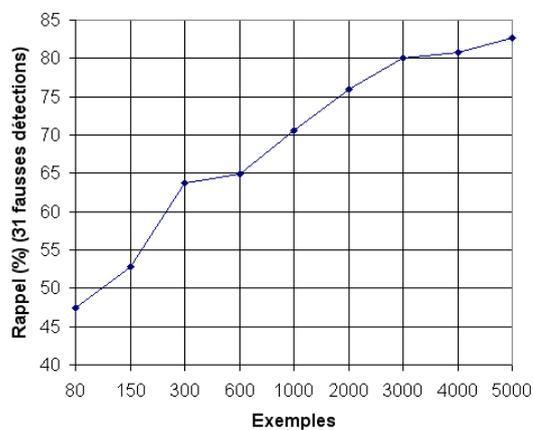


FIGURE 4.5 – Rappel du système de détection basé sur un MLP pour 31 fausses détections sur la base CMU en fonction du nombre d'exemples de la base de référence. On constate une augmentation du Rappel avec le nombre d'exemples.

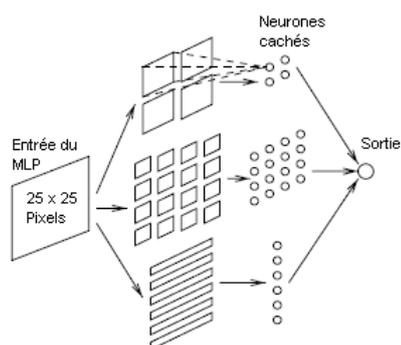
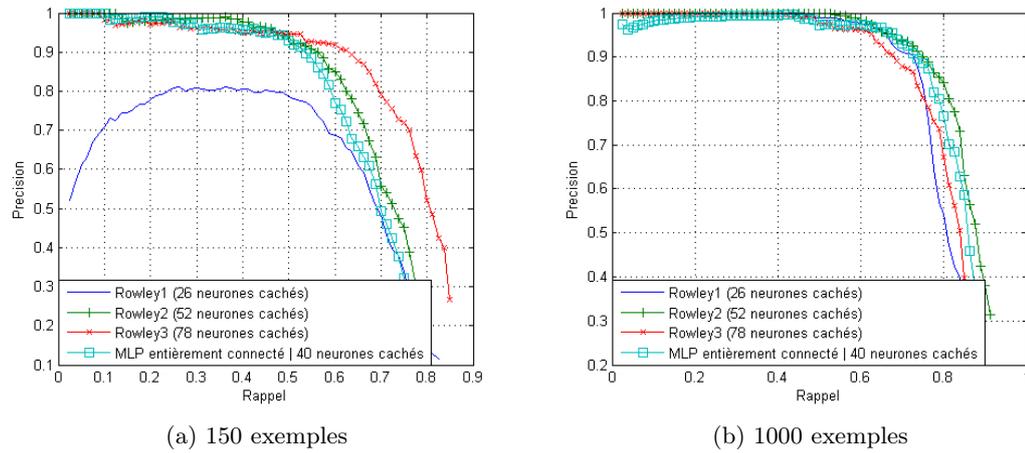
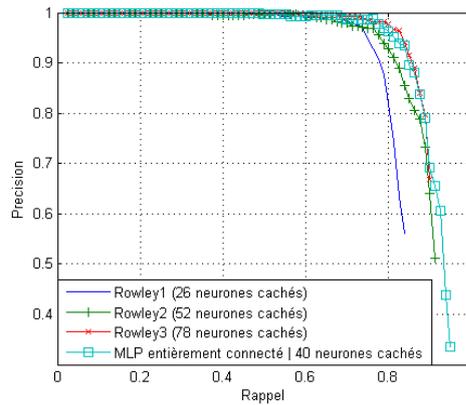


FIGURE 4.6 – Forme du MLP partiellement connecté utilisé, inspiré par Rowley *et al.*



(a) 150 exemples

(b) 1000 exemples



(c) 5000 exemples

FIGURE 4.7 – Influence du nombre de neurones cachés sur le réseau de neurones de Rowley *et al* avec 150 , 1000 et 5000 exemples d'apprentissage. Comparaison des résultats sur la base CMU par rapport à un MLP entièrement connecté à 40 neurones cachés. Nous pouvons constater que le réseau de neurone de Rowley est capable d'obtenir des résultats équivalents voire supérieurs a ceux du MLP entièrement connecté tout en possédant bien moins de connections.

4.3 C-PCA et association de MLP

L'utilisation d'un système d'apprentissage performant basé sur un MLP et la méthode d'Online BootStrapping permettent d'obtenir des résultats qui se rapprochent de l'état de l'art, à condition d'utiliser plusieurs milliers d'exemples. Ainsi, un tel système de détection, s'il s'avère bien plus performant que notre système basé sur la corrélation, nécessite l'utilisation de cinquante fois plus d'exemples d'apprentissage. Si nous utilisons 80 exemples, alors nous obtenons des résultats comparables à ceux obtenus par notre système basé sur l'association de corrélations. Le but de cette section est d'appliquer les méthodes employées avec la détection par corrélation au système de détection utilisant un MLP. Nous souhaitons ainsi obtenir un système capable d'atteindre des taux de détection au niveau de l'état de l'art mais utilisant une base d'apprentissage réduite. Nous avons constaté dans le chapitre précédent que l'association de plusieurs corrélations et de traitements d'image distincts permettait une nette amélioration des taux de détection. Dans cette section, nous décrivons de quelle manière nous avons transposé la méthode d'association de corrélations à l'association de MLP et en particulier comment nous avons adapté l'algorithme d'apprentissage à l'utilisation d'une association de MLP.

4.3.1 méthodes d'association des MLP

Nous avons dénombré trois méthodes simples d'association des classifieurs : la méthode 'ET', la méthode 'OU' et enfin la méthode 'vote'. Ces trois méthodes sont utilisées et comparées dans le système de détection de Rowley *et al.* En effet, afin de diminuer le nombre de fausses détections, ce système entraîne plusieurs MLP dont les résultats sont arbitrés de la façon suivante.

1. La méthode 'ET' consiste à ne garder que les résultats pour lesquels chacun des MLP donne un score supérieur à 0. Autrement dit, un visage est détecté s'il l'est pour l'ensemble des classifieurs.
2. La méthode 'OU' consiste à garder tous les résultats pour lesquels au moins un des classifieurs donne un résultat supérieur à 0. C'est à dire, garder toutes les détections qu'au moins un classifieur classe dans la catégorie 'visage'.
3. La méthode du 'vote' consiste à utiliser trois classifieurs et à classer une image comme 'visage', si au moins deux des trois classifieurs classent l'image dans la catégorie 'visage'. Cette méthode peu être considérée comme une solution intermédiaire entre le 'ET' et le 'OU'.

Dans le cas du système de Rowley *et al.*, l'entrée de chaque classifieur ainsi que les exemples de visages utilisés sont les mêmes. Chaque classifieur est entraîné indépendamment par une méthode de BootStrapping. Ainsi, les résultats de chaque classifieur ne se distinguent que par les variations des conditions initiales de chacun des MLP. Malgré cela, l'association de classifieurs permet une nette diminution du nombre des fausses détections, particulièrement pour les méthodes 'ET' et 'vote'. Un seul classifieur permet d'obtenir un taux de détection de 91,7% mais pour un total de

484 fausses détections avec la base de test CMU. Deux classifieurs similaires associés selon la méthode ‘ET’ permettent d’obtenir un Rappel de 86,6% avec 79 fausses détections. La méthode de ‘vote’ avec trois classifieurs permet d’atteindre 88,4% avec 99 fausses détections. La méthode ‘OU’ semble moins efficace et atteint un Rappel assez élevé de 90,3% mais pour un nombre assez important de 185 fausses détections.

Dans le chapitre précédent, nous avons associé les résultats de plusieurs corrélations selon une méthode basée sur le ‘ET’. En effet, pour qu’une détection soit valide, il est nécessaire que chaque corrélation de l’association renvoie un score supérieur à un seuil s_f (section 3.6.1). Nous avons fait ce choix pour plusieurs raisons :

- La première est que la probabilité que la rétine testée dans une image ne soit pas un objet est ‘infiniment’ plus grande que le contraire. Ainsi, un système de détection doit favoriser en priorité l’élimination des fausses détections. L’association selon la méthode ‘ET’ permet de limiter au maximum ces fausses détections puisque, si une seule corrélation de l’association renvoie un score indiquant que la rétine testée n’appartient pas à la catégorie ‘objet’, le système considère que l’image testée appartient alors à la catégorie ‘non objet’.
- La seconde est liée à la complexité de calcul. Comme nous l’avons vu (section 3.6.1) l’association par la méthode ‘ET’ permet de limiter au minimum le nombre de corrélations nécessaires à la détection. En effet, si une seule des corrélations de l’association renvoie un score inférieur au seuil s_f , il est inutile d’effectuer les autres corrélations de l’association.
- La troisième est liée à l’état de l’art. En effet, le système de détection d’objets le plus couramment utilisé actuellement, c’est à dire le détecteur de Viola-Jones [122] est basé sur une cascade de classifieurs dont le principe revient à associer une multitude de classifieurs simples selon la méthode ‘ET’ (un objet est détecté s’il est classé dans la catégorie ‘objet’ par l’ensemble des classifieurs de la cascade).

Dans cette section, nous nous inspirons du système de détection mis au point pour la corrélation pour adapter les méthodes d’association ‘ET’, ‘OU’ et ‘vote’ à notre système de détection utilisant un MLP. Le fonctionnement du système de détection utilisant l’association de MLP est décrit par la figure 4.8. Afin que chaque MLP de l’association utilise une information différente, l’image en Niveaux de Gris correspondant à la rétine dans l’image test est traitée de manière différente pour chaque MLP, puis normalisée. Les scores s_i de sortie de chaque MLP sont associés de façon à renvoyer un score s permettant de classer l’image correspondant à la rétine dans la catégorie appropriée. Plus le score s sera proche de 1 plus l’image correspondante aura de chance d’appartenir à la catégorie ‘objet’, à l’inverse, plus le score s’approche de -1 et plus l’image traitée par l’association de MLP aura de chance d’appartenir à la catégorie ‘non objet’. Contrairement au système de détection de Rowley *et al*, nous souhaitons obtenir, en sortie de l’association de classifieurs, un score de détection et non simplement l’appartenance à une des deux classe considérées. Pour ce faire, les méthodes ‘ET’, ‘OU’, et ‘vote’ sont adaptées comme suit :

1. La méthode ‘ET’ : le score s de classification correspond au score le plus faible

de l'ensemble des classifieurs, $s = \min(s_i)$.

2. La méthode 'OU' : le score s de classification correspond au score le plus élevé de l'ensemble des classifieurs, $s = \max(s_i)$.
3. La méthode 'vote' que nous nommerons '+' : le score s correspond à la valeur moyenne des scores s_i de l'ensemble des classifieurs, $s = \frac{1}{M} \sum_{i=1}^M s_i$. Cette méthode est un intermédiaire entre le 'ET' et le 'OU'.

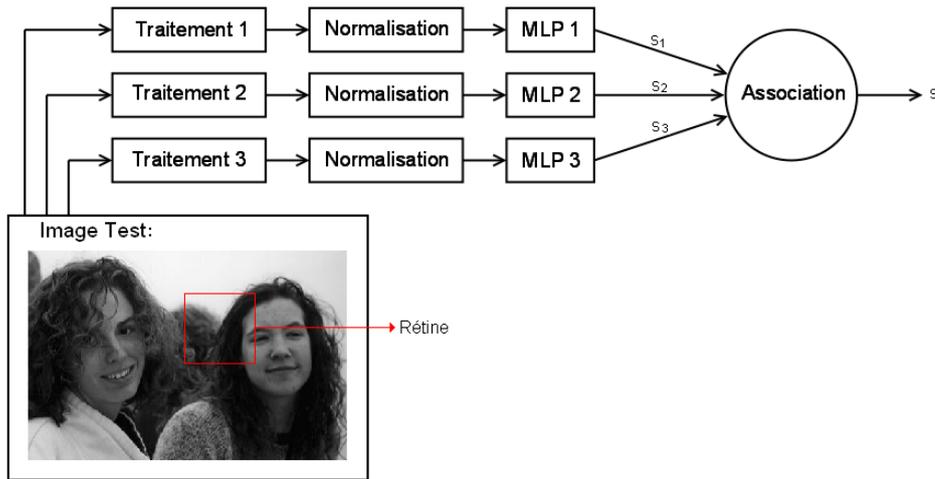


FIGURE 4.8 – Système de détection par association de MLP (exemple avec trois MLP). La rétine extraite de l'image de test est traitée selon trois méthodes donnant trois images distinctes qui sont ensuite normalisées. Le traitement d'image associé au MLP constitue alors une mesure de similarité dont le score correspondant est noté s_i . Les scores de similarité sont ensuite associés de façon à obtenir un score final s permettant de classer au mieux la rétine extraite de l'image test.

4.3.1.1 BootStrapping appliqué à l'association de MLP

Nous avons vu dans la section précédente, de quelle manière nous associons plusieurs MLP. Nous avons cependant passé sous silence la méthode utilisée pour entraîner les MLP de l'association. Nous proposons dans cette section, de décrire la méthode d'apprentissage de l'association de MLP. La méthode utilisée est une simple variation de l'algorithme d'Online BootStrapping (section : 4.2.1.1), ou plus exactement une adaptation de cette méthode à l'association de classifieurs. Elle permet d'effectuer l'apprentissage pour l'ensemble des MLP et non indépendamment pour chaque MLP. Pour ce faire, nous modifions l'algorithme d'Online BootStrapping comme suit :

N_{ftmax} : nombre maximum d'échantillons de 'non visage'.

N_{ft} : nombre total d'échantillons de 'non visage' utilisés pour l'apprentissage.

1. Réunir un petit nombre d'échantillons exemples de la classe 'non visage' et un ensemble d'échantillons représentatif de la classe 'visage'

2. Entraîner chacun des M MLP à partir de la base d'échantillons exemples courante par l'algorithme de Backpropagation.
3. Pour chacun des M MLP, supprimer les échantillons de 'non visage' dont le score de classification s_i est le plus proche de '-1' de façon à ce que $N_{ft} \leq N_{ftmax}$ pour chaque MLP.
4. Utiliser le système de détection sur une base d'images ne contenant pas la classe à détecter. Obtenir N_{fdmax} fausses détections supplémentaires.
5. Retourner à l'étape 2 jusqu'à ce que $FAR = \frac{N_{fd}}{N_{test}} < FAR_{min}$

Ainsi, chaque MLP est entraîné de la même façon que lorsque nous n'en utilisons qu'un. Cependant, les fausses détections sélectionnées correspondent aux fausses détections obtenues par l'association de classifieurs. De cette manière, le système se focalise sur les erreurs commises par l'association de classifieurs et non sur les erreurs commises par chaque MLP de l'association. De plus, chaque MLP conserve les N_{ftmax} échantillons de 'non visage' qu'il a le plus de difficultés à classer dans la catégorie adéquate. Ainsi, si les échantillons 'visage' sont les mêmes pour chaque MLP (à l'exception du traitement appliqué aux images), ce n'est pas le cas des échantillons de 'non visage'. De cette manière, chaque MLP se focalise sur les échantillons de 'non visage' qui entraînent des erreurs de classification.

4.3.1.2 Utilisation de la C-PCA

Afin que chaque MLP utilise une information différente, chaque imagerie présentée à chaque MLP subit un traitement d'image distinct. Nous avons vu dans le chapitre concernant la corrélation que la C-PCA permet de déterminer des filtres adaptés aux images à traiter. Ces filtres, nous ont permis d'obtenir une très nette amélioration des résultats par l'association des corrélations normées centrées des images filtrées par la C-PCA. Comme dans la section 3.6.1, nous proposons d'utiliser les filtres de la C-PCA afin d'effectuer les traitements d'image appliqués aux imageries d'entrée de chaque MLP, excepté pour le premier MLP qui utilise la totalité de l'information disponible, c'est à dire l'image en Niveaux de Gris.

Ainsi le 'traitement 1' consiste à garder l'image en Niveaux de Gris, le 'traitement 2' effectue une convolution entre le second filtre de la C-PCA et l'image en Niveaux de Gris, le 'traitement 3' effectue une convolution avec le troisième filtre de la C-PCA et ainsi de suite en fonction du nombre M de MLP.

Les filtres de la C-PCA de dimension 5×5 ayant montré de bons résultats avec la corrélation, nous conserverons cette dimension pour notre système basé sur les MLP. La figure 4.9 montre le résultat des traitements d'image correspondant aux trois premiers MLP appliqués à un visage en Niveaux de Gris. Nous pouvons constater que chacune des trois images obtenues est visuellement assez différente et permet de focaliser chaque MLP sur des informations distinctes.



FIGURE 4.9 – Exemples de traitement d’image utilisés pour le système de détection par l’association de MLP utilisant la C-PCA. La première image correspond à l’image en Niveaux de Gris, la seconde à l’image filtrée par le second filtre de la C-PCA et la troisième par le troisième filtre de la C-PCA.

4.3.2 Résultats expérimentaux

Nous proposons dans cette section, de mesurer l’efficacité et l’utilité du système de détection par association de MLP tel que nous l’avons construit et en particulier, les performances de ce système avec une base d’exemples d’apprentissage réduite. Pour ce faire, nous utiliserons toujours la base de test CMU et les exemples tirés de la base ‘Caltech WebFaces’. Le MLP partiellement connecté ‘Rowley2’ comportant 52 neurones cachés ayant montré les meilleurs compromis performance temps de calculs, nous utiliserons toujours ce dernier dans les expérimentations suivantes, excepté si nous précisons le contraire.

Nous commencerons par mesurer l’influence de l’utilisation des filtres de la C-PCA sur le système de détection. Nous observerons ensuite l’influence de la méthode d’association utilisée (‘ET’, ‘OU’, ‘+’) ainsi que du nombre de MLP. Finalement, nous utiliserons la méthode d’association la plus performante et mesurerons l’influence du nombre N d’exemples d’apprentissage. Nous comparerons ensuite ces résultats à ceux obtenus avec un seul MLP puis avec l’état de l’art.

4.3.2.1 Influence des filtres de la C-PCA

Nous avons, jusqu’à présent, mesuré l’efficacité du système de détection utilisant l’image en Niveaux de Gris normalisée comme ‘entrée’ d’un MLP. Avant de s’intéresser aux résultats de l’association de MLP utilisant les images traitées par convolution avec les filtres de la C-PCA, nous avons appliqué chaque mesure de similarité de notre système à la base de test CMU afin de pouvoir convenablement mesurer l’influence de l’association des MLP. Comme nous souhaitons que notre système soit fonctionnel avec peu d’exemples, nous effectuons ces mesures avec 150 exemples d’apprentissage. La figure 4.10 montre les courbes Rappel Précision obtenues avec les seconds, troisièmes et quatrièmes filtres de la CPA correspondant respectivement aux ‘traitements 2, 3 et 4’. Nous avons ajouté les résultats avec l’image en Niveaux de Gris correspondant au ‘traitement 1’ comme point de comparaison. Afin de donner des résultats chiffrés, nous donnons aussi dans le tableau 4.2, le Rappel du système de détection en fonction du traitement d’image appliqué pour 31 et 65 fausses détections.

Le ‘traitement 3’ semble donner les meilleurs résultats alors que les taux de détection diminuent légèrement avec l’utilisation des filtres 2 et surtout 4 de la C-PCA. Cependant, les résultats obtenus restent du même ordre, quel que soit le traitement appliqué ; nous pouvons ainsi espérer que l’association de ces traitements d’image permettra une amélioration des résultats, aucun des traitements utilisés ne risquant de faire chuter les taux de détection. De même, lorsque l’on observe le nombre de fichiers *NbIm* nécessaire à la convergence de l’algorithme de BootStrapping, il reste du même ordre quel que soit le traitement appliqué, c’est à dire entre 100 et 200.

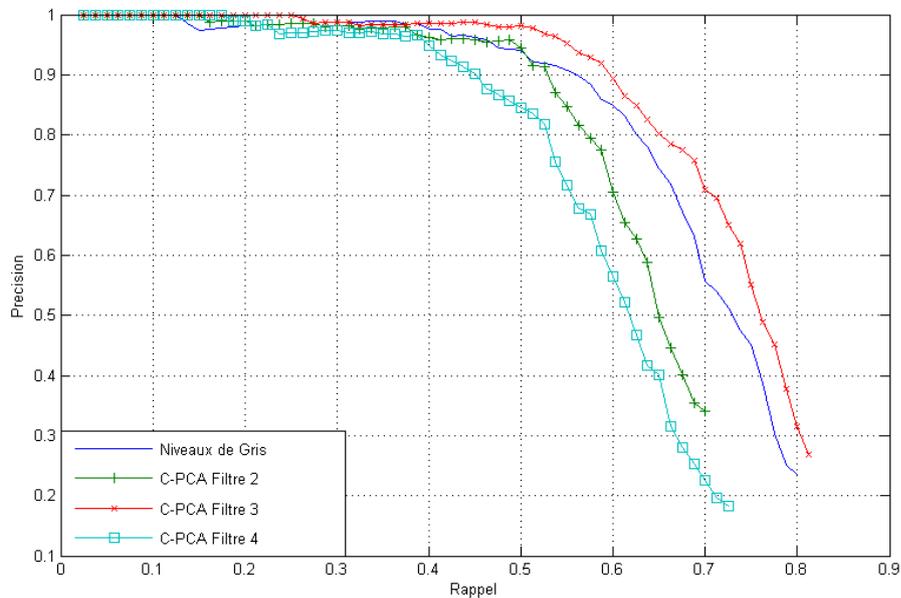


FIGURE 4.10 – Courbes Rappel Précision du système de détection basé sur le MLP Rowley2. Comparaison des résultats en fonction du traitement d’image appliqué aux images classées par le MLP. Nous pouvons constater que les résultats restent comparables, que l’on utilise l’image en Niveaux de Gris, ou les filtres de la C-PCA

fausses détections	Traitement			
	1	2	3	4
31	55,0%	51,8%	58,1	44,7%
65	60,1%	55,4%	62,3%	51,4%

TABLE 4.2 – Comparaison du Rappel du système de détection basé sur un MLP en fonction du traitement d’image appliquée pour 31 et 65 fausses détections sur la base de test CMU.

4.3.2.2 Influence de la méthode d'association et du nombre de MLP

Comme nous l'avons vu section 4.3.1, nous pouvons associer les mesures de similarité selon trois différentes méthodes, *i.e.*, 'ET', 'OU' et '+'. De plus, nous pouvons varier le nombre de mesures de similarité utilisées. Le nombre de combinaisons étant alors assez important, nous commencerons par observer l'influence du nombre M de mesures de similarité pour chacune des trois associations. Une fois déterminé le nombre M optimal, nous comparerons alors les résultats de chacune des trois associations. Nous utilisons toujours 150 exemples d'apprentissages et évaluons notre système sur la base de test CMU.

Méthode 'ET' : La figure 4.11 présente les courbes Rappel Précision pour une seule mesure de similarité utilisant l'image en Niveaux de Gris ($M = 1$), puis pour deux mesures combinant l'image en Niveaux de Gris et le second filtre de la C-PCA ($M = 2$) jusqu'à combiner quatre mesures de similarité ($M = 4$) associant ainsi, l'image en Niveaux de Gris et trois filtres de la C-PCA par la méthode 'ET'. Nous pouvons, en premier lieu, constater que l'ajout d'une mesure de similarité supplémentaire ne détériore pas les résultats et permet à partir de $M = 3$ une sensible amélioration des résultats. Le tableau 4.3 reporte le Rappel pour 31 et 65 fausses détections en fonction de M et montre que l'association de trois mesures de similarité permet, pour un même nombre de fausses détections, une augmentation du Rappel de l'ordre de 5% par rapport à l'utilisation d'une seule mesure de similarité.

Si le gain de performances apporté par l'association 'ET' est relativement faible, le nombre d'images nécessaire $NbIm$ à la convergence de l'algorithme de BootStrapping n'est que de 25 pour $M \geq 2$ contre 137 pour $M = 1$. Ceci permet une diminution appréciable de la durée de la phase d'apprentissage.

fausses détections	nombre d'associations M			
	1	2	3	4
31	55,0%	53,0%	61,3	61,5%
65	60,1%	60,5%	65,7%	65,5%

TABLE 4.3 – Comparaison du Rappel du système de détection basé sur les MLP en fonction du nombre M de mesures de similarité utilisées avec la méthode d'association 'ET' pour 31 et 65 fausses détections sur la base de test CMU.

Méthode 'OU' : Comme pour la méthode 'ET', nous reportons les résultats obtenus en fonction de M , sous forme de courbes Rappel Précision (figure : 4.12) et de tableau (tableau : 4.4). Comme nous le pensions, la méthode 'OU' améliore le Rappel au détriment de la Précision. En effet, nous pouvons constater sur la figure 4.12 que la Précision du système de détection pour un Rappel inférieur à 0.5 a tendance à légèrement diminuer avec l'augmentation du nombre M d'associations. A l'inverse, nous constatons que le gain sur le Rappel apporté par l'association 'OU' est d'autant plus important que la Précision diminue. Ainsi, pour $M = 3$ le Rappel est augmenté

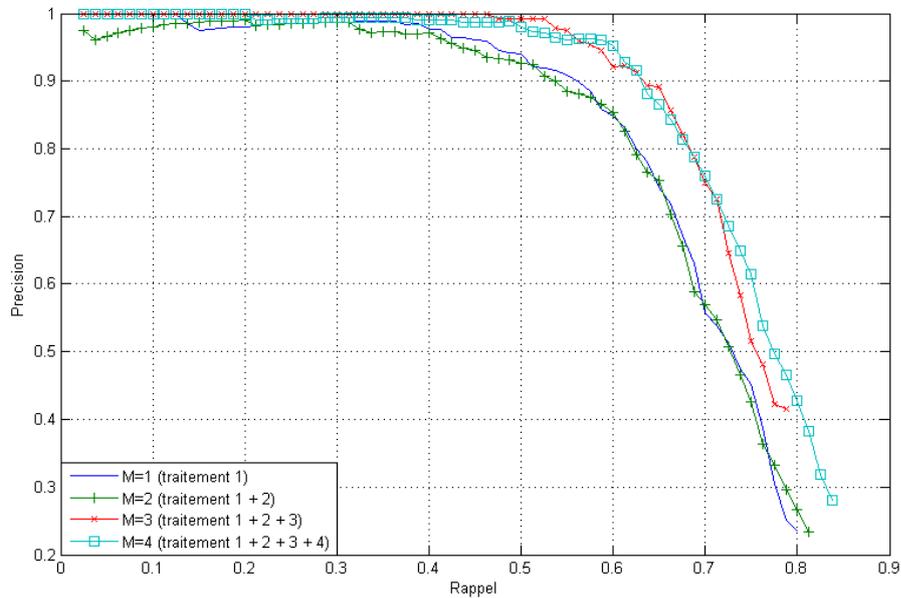


FIGURE 4.11 – Courbes Rappel Précision du système de détection basé sur l'association de MLP 'ET' et la C-PCA. Comparaison des résultats en fonction de M , le nombre de mesures de similarité utilisées.

de 10% pour 31 fausses détections et de 15% pour 65. Comme pour l'association 'ET' l'utilisation d'une quatrième mesure de similarité ne semble pas permettre de réellement améliorer les résultats. Ainsi, l'utilisation de $M = 3$ associations semble le meilleur compromis entre la complexité du système et les performances.

Le nombre de fichiers images $NbIm$ nécessaire à l'apprentissage est lui nettement augmenté dès que $M \geq 2$ puisqu'il est de l'ordre de 450 images, contre 137 pour $M = 1$ et 25 en utilisant la méthode 'ET'.

fausses détections	nombre d'associations M			
	1	2	3	4
31	55,0%	63,9%	65,5	60,3%
65	60,1%	69,4%	72,2%	70,0%

TABLE 4.4 – Comparaison du Rappel du système de détection basé sur les MLP en fonction du nombre M de mesures de similarité utilisées avec la méthode d'association 'OU' pour 31 et 65 fausses détections sur la base de test CMU.

Méthode '+' : Cette méthode constitue un compromis entre la méthode 'ET' qui favorise la Précision du système en minimisant le nombre de fausses détections et

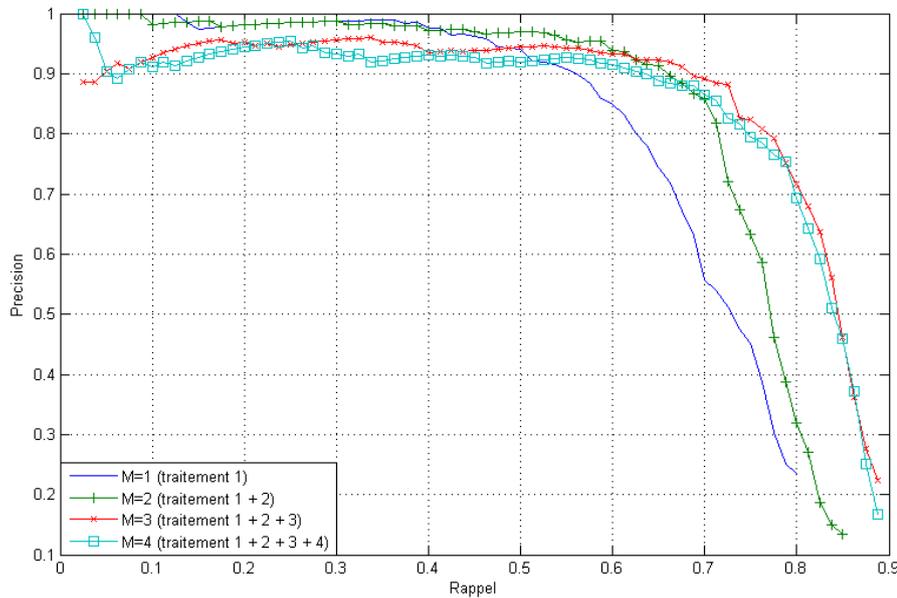


FIGURE 4.12 – Courbes Rappel Précision du système de détection basé sur l’association de MLP ‘OU’ et la C-PCA. Comparaison des résultats en fonction de M , le nombre de mesures de similarité utilisées.

la méthode ‘OU’ qui au contraire, favorise l’augmentation du Rappel du système de détection. Comme pour les deux autres méthodes, les résultats obtenus sur la base CMU sont reportés sous forme de courbes Rappel Précision (figure : 4.13) et sous forme de tableau (tableau : 4.5). Nous pouvons constater que l’association ‘+’ permet une nette amélioration des taux de détection avec un gain de l’ordre de 20% du Rappel pour $M = 3$ et 31 et 65 fausses détections. Comme pour les méthodes d’association précédentes, l’augmentation du nombre d’associations M au delà de 3 ne semble pas apporter un gain de performance le justifiant. Ainsi, nous remarquons que pour les trois méthodes d’association ainsi que pour la méthode de détection par association de corrélations (section 3.6.1), l’association de trois mesures de similarité est le meilleur compromis entre les taux de détection du système et les temps de calculs.

La méthode d’association ‘+’ n’apporte par contre aucun gain particulier sur la convergence de l’algorithme de BootStrapping, $NbIm$ restant de l’ordre de 150 quel que soit la valeur de M .

Comparaison des méthodes ‘ET’, ‘OU’ et ‘+’ : Les expérimentations précédentes nous ont permis de déterminer que le meilleur nombre d’associations est $M = 3$ pour les trois méthodes utilisées. Afin de déterminer la méthode d’association la plus performante nous reportons (figure : 4.14) les courbes Rappel Précision des trois méthodes ‘ET’, ‘OU’ et ‘+’ pour $M = 3$, nous avons en plus reporté la

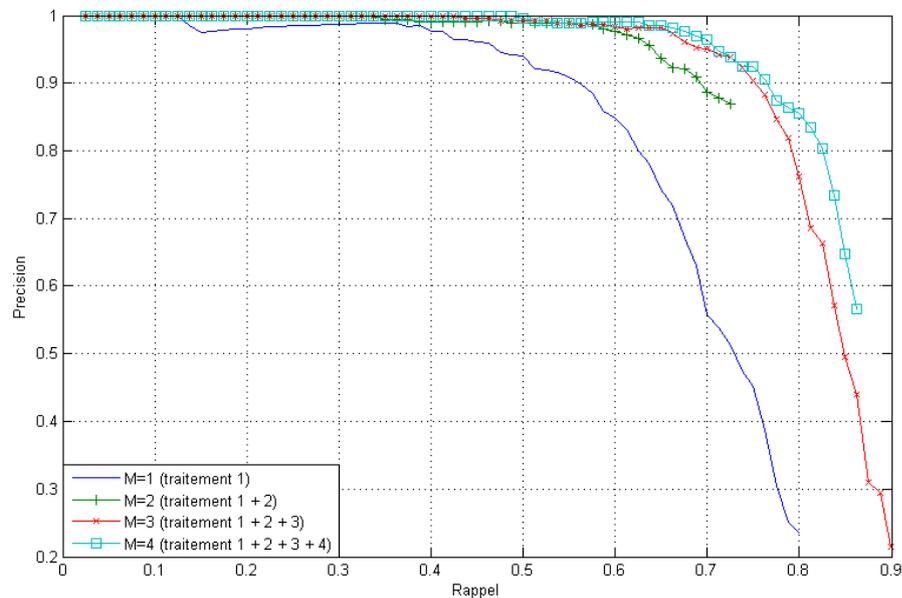


FIGURE 4.13 – Courbes Rappel Précision du système de détection basé sur l’association de MLP ‘+’ et la C-PCA. Comparaison des résultats en fonction de M , le nombre de mesures de similarité utilisées.

fausses détections	nombre d’associations M			
	1	2	3	4
31	55,0%	66,7%	72,6	73,8%
65	60,1%	73,0%	76,2%	78,6%

TABLE 4.5 – Comparaison du Rappel du système de détection basé sur les MLP en fonction du nombre M de mesures de similarité utilisées avec la méthode d’association ‘+’ pour 31 et 65 fausses détections sur la base de test CMU.

courbe Rappel Précision correspondant au système de détection sans association afin de mieux visualiser le gain apporté par la méthode d’association de MLP utilisant la C-PCA. Nous pouvons aisément constater que, quelle que soit la méthode d’association utilisée, nous obtenons un gain substantiel des taux de détection. De plus, nous voyons que la méthode ‘+’ permet de tirer partie des avantages de la méthode ‘ET’ qui favorise la Précision et de la méthode ‘OU’ qui favorise le Rappel. Ainsi, la méthode d’association ‘+’ de trois MLP partiellement connectés (Rowley2) permet d’obtenir un système de détection efficace avec seulement 150 visages exemples contre de l’ordre de 4000 pour l’état de l’art.

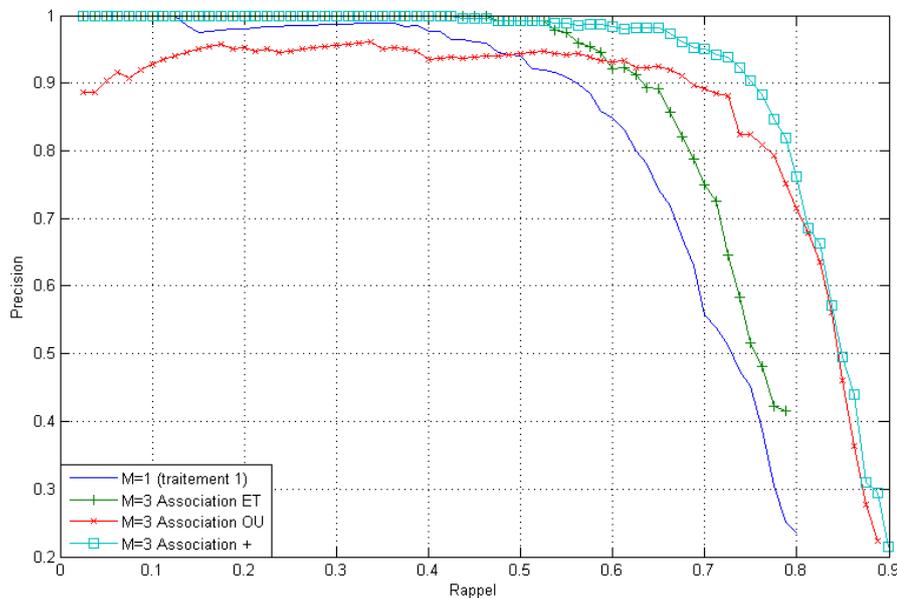


FIGURE 4.14 – Courbes Rappel Précision du système de détection basé sur l’association de MLP et la C-PCA. Comparaison des résultats en fonction de la méthode d’association utilisée.

4.3.2.3 Influence du nombre d’exemples

Les expérimentations précédentes nous ont permis de mettre au point un système de détection capable de détecter des visages dans une base complexe avec une base d’apprentissage réduite à seulement 150 exemples. Ce système consiste à utiliser une association ‘+’ de trois MLP partiellement connectés (Rowley2) avec l’image en Niveaux de Gris et les filtres 2 et 3 de la C-PCA. Cette méthode permet, à partir de seulement 150 exemples d’apprentissage, d’atteindre un Rappel de 72,6% pour 31 fausses détections. Ces résultats restent néanmoins assez éloignés des taux de détection de l’état de l’art qui sont au moins de 85% pour le même nombre de fausses détections. Afin de tester dans quelle mesure notre système est sensible aux variations du nombre N d’exemples d’apprentissage et combien d’exemples sont nécessaires pour obtenir des résultats comparables à l’état de l’art, nous avons évalué notre système sur la base de test CMU avec $N = 80, 150, 300, 600$ et 1000 exemples. Les courbes Rappel Précision sont données figure 4.15. Nous pouvons constater une amélioration rapide des résultats avec l’augmentation du nombre d’exemples jusqu’à 600, puis une stagnation, voire une très légère régression avec 1000 exemples d’apprentissage. Ceci s’explique par le fait que nous avons dimensionné notre système pour fonctionner avec peu d’exemples.

La figure 4.16 compare le Rappel obtenu par l’association de MLP pour 31 fausses détections à celui obtenu sans association (section : 4.2.2.3). Nous remarquons un

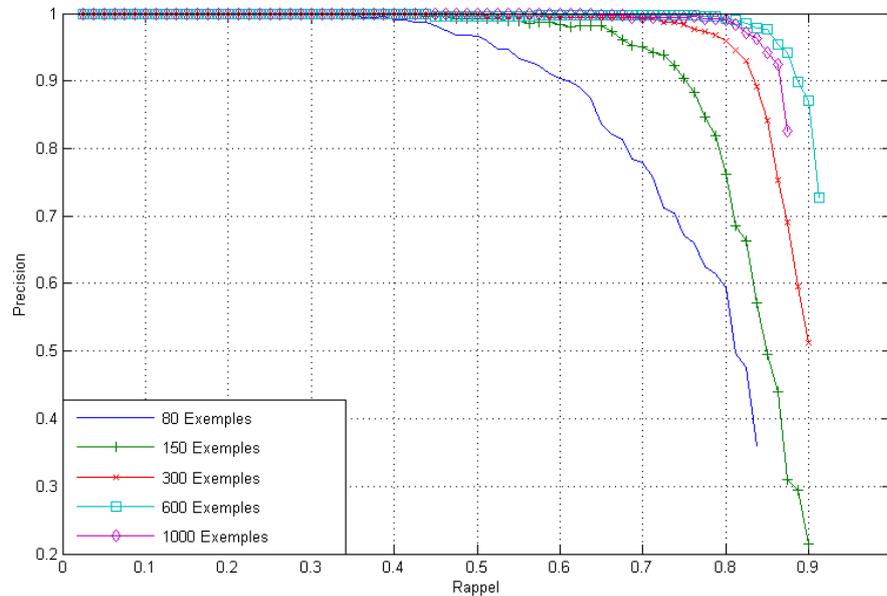


FIGURE 4.15 – Courbes Rappel Précision du système de détection basé sur l'association '+' de MLP et la C-PCA. Comparaison des résultats en fonction du nombre d'exemples d'apprentissage.

écart de près de 20% lorsque nous utilisons moins de 1000 exemples d'apprentissage. Nous pouvons aussi constater que l'association de classifieurs permet d'atteindre avec 300 exemples les mêmes résultats qu'un seul MLP avec 5000 exemples (figure : 4.4 et 4.5). Nous démontrons ainsi que l'utilisation de filtres détecteurs de contours et d'associations de classifieurs permet, comme pour la corrélation, une très nette amélioration des résultats lorsque nous utilisons peu d'exemples.

Enfin, si l'on se compare à l'état de l'art (tableau : 4.6), nous constatons que notre système atteint des performances du même ordre que le système de Viola-jones ou celui de Rowley *et al* avec seulement 600 exemples d'apprentissage contre plusieurs milliers pour ces derniers. Nous constatons de plus que les meilleurs systèmes en terme de taux de détections sont ceux utilisant le plus grand nombre d'exemples d'apprentissage.

Afin d'illustrer les résultats de notre système de détection avec 600 exemples, nous montrons figure 4.17 quelques détections effectuées sur la base CMU. Nous pouvons ainsi remarquer que notre système de détection échoue à détecter les visages très peu contrastés ou très flous mais est capable de détecter des visages subissant tout de même de grandes variations d'illumination ou de forme.

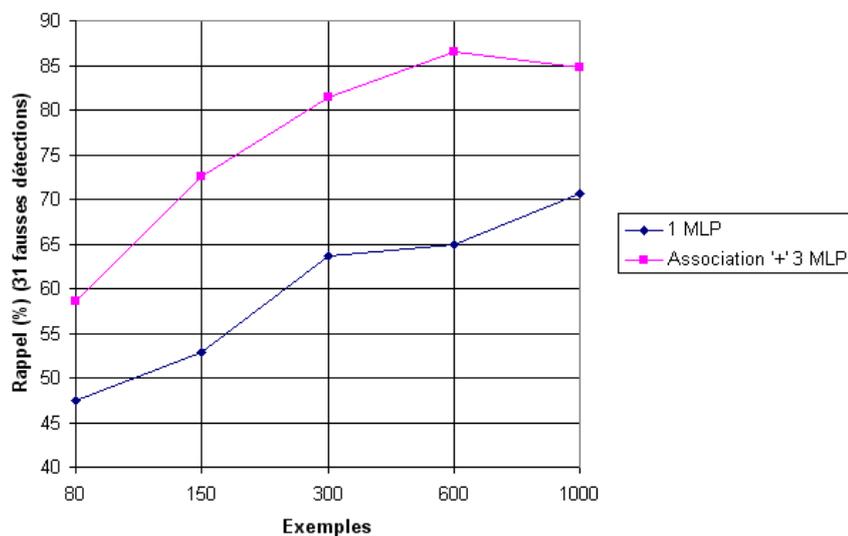


FIGURE 4.16 – Rappel du système de détection basé sur un MLP et l’association ‘+’ de trois MLP pour 31 fausses détections sur la base CMU en fonction du nombre d’exemples de la base de référence.

Système de détection	Fausses détections				
	0	10	31	65	167
Rowley <i>et al</i> [42] (4000)	-	83,2%	86,0%	-	90,1%
Viola-Jones [122] (4916)	-	78,3%	85,2%	89,8%	91,8%
CFF [39] (3702)	88,8%	90,5%	91,5%	92,3%	93,1%
Association ‘+’ 3 MLP (300)	49,6%	76,2%	81,4%	82,9%	85,5%
Association ‘+’ 3 MLP (600)	43,1%	84,0%	86,5%	88,3%	89,7%

TABLE 4.6 – Rappel du système de détection basé sur l’association ‘+’ de trois MLP et la C-PCA pour divers nombres de fausses détections sur la base de test CMU. Les nombres entre parenthèses correspondent au nombre d’exemples de la base d’apprentissage.



FIGURE 4.17 – Exemples de détections effectuées sur la base CMU par le système basé sur l'association '+' de trois MLP avec 600 exemples d'apprentissage.