

Voici mon compte-rendu sur les deux exercices du premier cours (indices bas niveau). N'hésitez pas à me contacter à la moindre question ! Merci d'avance pour votre lecture.

Exercice 0.1 Descripteurs issus des réseaux de neurones

1 - On lit dans la description du modèle AlexNet la taille de l'activation de la couche fc7 : 4096. Le tableau activ quant à lui a une taille (size) de 1 1 4096.

2 - **Pourquoi le cosinus entre les deux descripteurs est-il une métrique adaptée pour mesurer la ressemblance ?** Chaque descripteur est un vecteur décrivant l'image. Lorsque l'on fait l'embedding de mots par exemple, la similarité cosinus (*cosine similarity*) est une technique usuelle pour calculer la similarité entre deux documents ou ensemble de mots par exemple.

De plus, dans notre cas, on se rend compte que si deux images sont "colinéaires" ($\text{imA} = k * \text{imB}$), alors les deux images sont en fait les mêmes, à un facteur multiplicatif près, c'est-à-dire à des luminosités/intensités lumineuses différentes. Autrement dit, si le cosinus entre leurs deux descripteurs est de 1, alors cela revient à dire que les deux images sont colinéaires (produit scalaire = produit des normes). La similarité cosinus étant insensible aux variations d'intensité lumineuse, c'est une bonne mesure de la ressemblance entre deux images et donc une bonne métrique pour l'utilisation qu'on en fait dans cet exercice (plus des images sont "colinéaires", plus elles doivent se ressembler).

3 - Analyser la pertinence des résultats.

On juge la qualité des résultats renvoyés par la fonction `plusProchesImages` en regardant combien d'images trouvées correspondent effectivement à l'image d'origine. Dans chacune des figures, on affiche les 9 images les plus semblables trouvées, en incluant l'image d'origine pour avoir une référence. On donne aussi les scores de ressemblance associés afin de discuter de la possibilité de fixer un seuil sur la valeur du descripteur, pour sélectionner *uniquement les bonnes images*, automatiquement.



Figure 1 – Résultats obtenus avec `plusProchesImages` pour `usine6`. Scores de similarité associés (ordre décroissant) : [1.0000, 0.9556, 0.9174, 0.8860, 0.8717, 0.8202, 0.8148, 0.8076, 0.7808]

- Pour la première image (`usine6`), il se trouve que les 7 premières images (sur 9) correspondent effectivement à l'image d'origine. Toutes ces images ont un score de similarité ≥ 0.81 . Il est

aussi intéressant de noter que certaines images présentes dans la base ne sont pas renvoyées par l'algorithme (par exemple, usine62 correspond à usine6 mais ne fait pas partie des images renvoyées, alors qu'elle serait clairement plus pertinente que la 9e image renvoyée dans la figure 1).

- Pour la deuxième image (usine8), toutes les images trouvées correspondent effectivement à l'image d'origine. Toutes ces images ont un score de similarité ≥ 0.71 .
- Pour la troisième image (usine40), seules les 2 premières images trouvées correspondent à l'image d'origine (alors qu'au moins une autre image non trouvée, usine38, était pertinente. Mais après vérification son "score cosinus" est beaucoup moins bon, de 0.6522). Leur score est ≥ 0.78 .

Globalement la technique utilisée renvoie des images pour la plupart très ressemblantes à l'image de base. Les premières (celles avec le meilleur score) correspondent souvent tout à fait à la scène originale, d'autres avec un score moindre correspondent à des scènes où certains éléments (tuyaux par exemple) sont communs à l'image d'origine, même si ce ne sont pas les mêmes éléments que l'image d'origine. Ainsi, le jugement de la qualité des images renvoyées dépend de l'application que l'on souhaite en faire ; s'il faut trouver des images partageant un maximum de caractéristiques, ou si l'on préfère se concentrer sur trouver toutes les images d'une seule et unique scène. Dans les deux cas, on obtient des résultats assez satisfaisants. Voir les figures 1, 2 et 3 pour analyser les résultats.



Figure 2 – Résultats obtenus avec `plusProchesImages` pour `usine8`. Scores de similarité associés (ordre décroissant) : [1.0000, 0.7797, 0.7672, 0.7580, 0.7440, 0.7379, 0.7274, 0.7183, 0.7104]

Au vu de ces résultats, on imagine pouvoir définir un seuil permettant de ne sélectionner que les images qui correspondent bien à la scène, **pour une scène donnée**. On voit en effet que pour chacun des 3 exemples présentés ici, une fois passée une certaine valeur seuil (par exemple en première approximation 0.81 pour `usine6`), les images renvoyées ne correspondent plus à la scène originale. Pour chaque scène, on peut ainsi imaginer définir un seuil. Cette méthode ne permet cependant pas d'être exhaustif, comme on le voit avec `usine40` pour laquelle `usine38` serait en-dessous du seuil, malgré sa pertinence.

En revanche, on ne peut pas vraiment définir de seuil "universel" pour automatiquement récupérer les images identiques à une scène, *quelle que soit la scène*. En effet, on voit encore sur ces trois exemples



Figure 3 – Résultats obtenus avec plusProchesImages pour usine40. Scores de similarité associés (ordre décroissant) : [1.0000, 0.7854, 0.7424, 0.7164, 0.7057, 0.7032, 0.7019, 0.6987, 0.6944]

que le seuil imaginable pour usine6 n'est pas compatible avec celui pour usine8 (on ne garderait pour usine8 que la première image - l'image originale -, ce qui est manifestement incorrect vu que les 8 autres images renvoyées correspondent bien à la scène). On peut imaginer, si l'on veut, garder comme seuil le max de tous les seuils rencontrés, mais ça n'est pas vraiment un critère satisfaisant.

4 - Comparez les résultats des deux expériences obtenus avec les couches fc7 et fc8.

Pour usine6 et usine8, les performances du descripteur fc8 semblent sensiblement moins bonnes que la couche fc7. En effet, pour usine6 par exemple, la 8e image pour fc7— devient la 6e pour fc8 (voir figure 4), alors qu'elle n'est pas vraiment très satisfaisante au vu de la scène originale. Constat similaire pour usine6 où une image contenant un boîtier blanc (mais pas le même que celui de l'image d'origine) apparaît en 7e position avec fc8 alors qu'elle était absente pour fc7. Pour usine40, les résultats sont sensiblement les mêmes (de toute manière, il y a peu d'images correspondant à la scène de usine40 dans la base d'images originale, ce qui se passe dans les deux dernières rangées n'est donc pas primordial).

On aurait pu imaginer que les descripteurs des couches plus profondes seraient plus à même de décrire fidèlement les images (d'après les cours que l'on a eus et la documentation Matlab, plus exactement, les premières couches apprennent les caractéristiques - *features* - "grossières" comme les contours, et les couches plus profondes apprennent des caractéristiques plus complexes), mais il semblerait que ça n'est pas forcément le cas.

Exercice 0.2 Mise en correspondance par corrélation

La figure 7 reproduit l'affichage de la figure 1 du TD.

Les résultats que l'on obtient en prenant $N = 25$ points, et $w = 2, 5, 15, 25$ sont affichés figure 8.

1 - Quelle est l'évolution de la carte de corrélation avec w ?

On visualise la carte de corrélation comme une surface à la figure 9 (attention aux couleurs, l'échelle n'est pas la même entre chaque image). Au fur et à mesure que w croît, les points les plus corrélés sont de plus en plus regroupés. Si pour $w = 3$ les points les plus corrélés sont assez dispersés au sein de l'image,



Figure 4 – A gauche : 9 meilleures images obtenues avec fc8 pour usine6. A droite : 9 meilleures images obtenues avec fc7.



Figure 5 – A gauche : 9 meilleures images obtenues avec fc8 pour usine8. A droite : 9 meilleures images obtenues avec fc7.

ils se "regroupent" progressivement autour de "zones" lorsque w augmente ; pour $w = 25$, on remarque que les points détectés peuvent être grossièrement découpés en 3 "zones" ; une autour du "bon" point, une autre autour d'une autre fenêtre adjacente de la maison, et une autour d'un point dans le ciel.

Au niveau de la carte de corrélation (figure 9), on remarque que l'augmentation de w amène à un certain "lissage" de cette dernière. Si pour $w = 3$ de nombreux pics se retrouvent au travers de toute l'image, au fur et à mesure que w augmente, ces pics sont moins forts et surtout moins nombreux. Pour $w = 25$, on trouve toujours des pics de corrélation, mais moins nombreux et plus lissés avec des "zones fortes" qui correspondent à priori aux 3 zones où se regroupent les points, que l'on a évoquées au paragraphe précédent.

2 - Pourquoi la valeur maximale ne correspond pas au point correspondant pour $w = 3$?

Pour $w = 3$, la fenêtre (le *template* de mise en correspondance) est la plus petite parmi les 4 testées. Plus cette fenêtre est petite, et moins on prend en compte le voisinage "éloigné" du point étudié. Ainsi, si le voisinage direct du point est relativement uniforme (contour d'une fenêtre, partie d'un mur uniforme en couleur) en termes de couleur et luminosité, et que de nombreux points ont un voisinage similaire dans l'image globale, alors une petite fenêtre de corrélation risque de donner lieu à de nombreux points candidats, et éventuellement à une mauvaise mise en correspondance. Ceci pourrait expliquer pourquoi la valeur maximale ne correspond pas au point correspondant pour $w = 3$.



Figure 6 – A gauche : 9 meilleures images obtenues avec fc8 pour usine40. A droite : 9 meilleures images obtenues avec fc7.

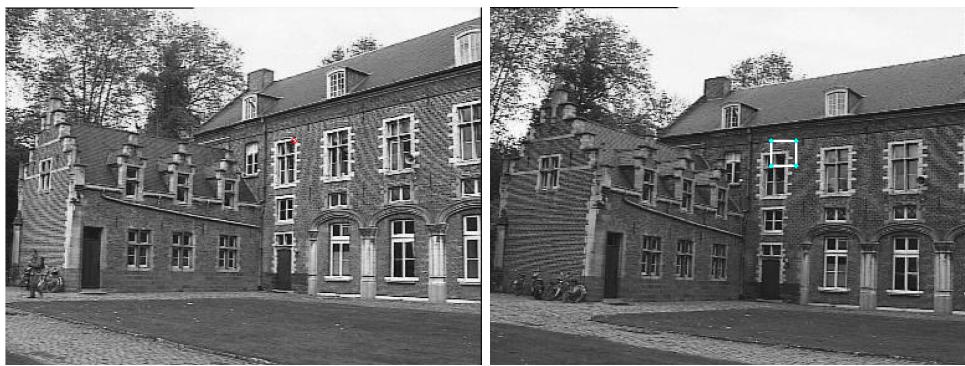


Figure 7 – Affichage pour la première question : point à mettre en correspondance dans l1 (à gauche), et le point avec la corrélation maximum dans l2 (à droite), pour $w=10$.

3 - Pourquoi les N points avec la plus grande corrélation sont-ils éparpillés quand w est petit, et concentrés en général au même point quand w croît ?

Quand w est petit, du fait de la petite taille de la fenêtre, comme on vient de le souligner, il est possible qu'un point diamétralement opposé au point d'origine ait un voisinage similaire, donnant lieu à de nombreux points dispersés. Cela est cohérent avec la forme de la carte de corrélation (figure 9), qui contient de nombreux pics répartis sur toute l'image. Lorsque w croît, au contraire, on prend de plus en plus le voisinage "éloigné" des points en compte, ainsi les valeurs de fortes corrélations se regroupent autour de zones similaires au point d'origine (voir réponse à la 1e question). Ceci explique que l'on ait un regroupement des N points avec la plus grande corrélation.

4 - Pourquoi y a-t-il parfois des points qui se promènent dans le ciel et induisent une corrélation forte ?

Il n'est pas évident d'avoir une certitude exacte permettant d'expliquer ce phénomène. La première chose que l'on peut se dire en voyant que des points dans le ciel donnent une corrélation forte est de supposer que le point considéré a effectivement une certaines ressemblance avec les points dans le ciel.

Cependant, lorsque l'on ré-itère l'algorithme sur un autre point aléatoire de l'image d'origine, on fait la même observation. A priori, cela ne dépend pas du point d'origine mis en correspondance (même si le point est en zone "non homogène" - frontière entre deux zones de couleurs différentes par exemple - le constat est le même). On voit cela sur la figure 10 (j'ai testé sur 3 points différents, toujours avec le même résultat).

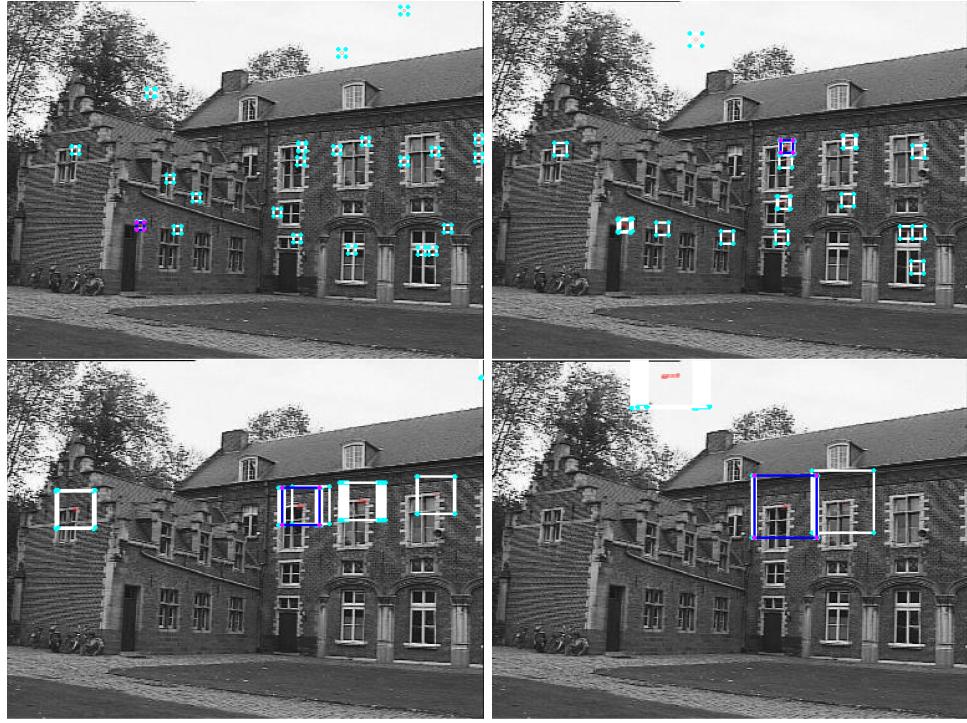


Figure 8 – Points trouvés par `find_N_highest_corr.m`. $N = 25$ pour chaque image. De gauche à droite, de haut en bas : $w = 3$, $w = 5$, $w = 15$, et $w = 25$.

Alors, la mise en correspondance de points du ciel avec un point correspondant à un bord de fenêtre peut être lié à :

- une similarité niveau couleur entre le rebord blanc de la fenêtre et le ciel gris (peu probable puisque même en testant différents points le problème subsiste)
- on a vu que ce phénomène se renforce lorsque la taille de la fenêtre (w) augmente. Peut-être que les points du ciel, d'un gris uniforme, ne sont pas intéressants pour w petit, mais puisque de couleur "moyenne" en niveau de gris, ils deviennent de plus en plus intéressant au fur et à mesure que des hétérogénéités dans le reste de l'image sont de moins en moins bien classées par rapport à "juste" le ciel gris. Je ne suis néanmoins pas certain de cette théorie.

En tout cas, ma curiosité est piquée par cette observation et je suis curieux de savoir pourquoi ces points se "baladent" dans le ciel.

Merci pour votre lecture. N'hésitez pas à me contacter à la moindre question.

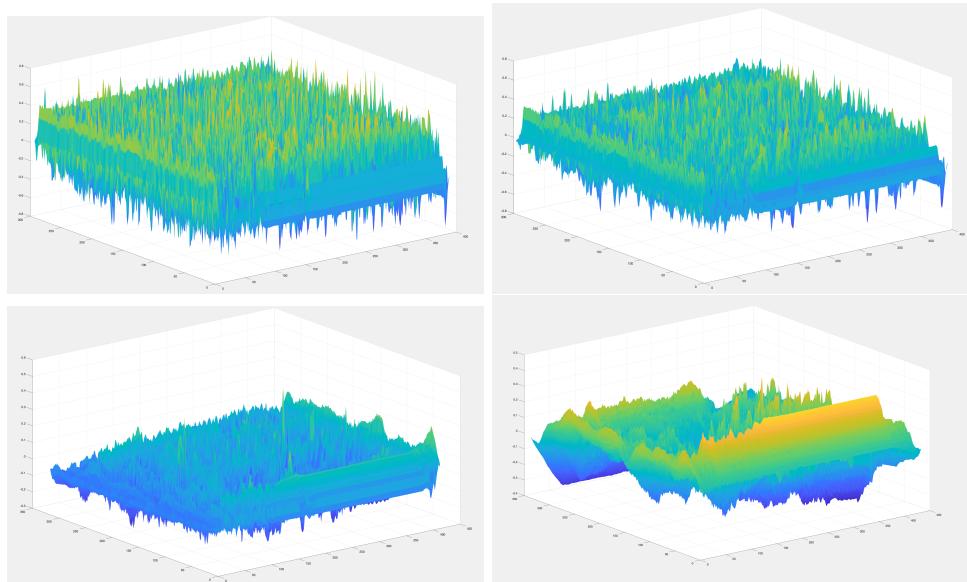


Figure 9 – Cartes de corrélation. $N = 25$ pour chaque image. De gauche à droite, de haut en bas : $w = 3$, $w = 5$, $w = 15$, et $w = 25$.



Figure 10 – Mise en correspondance par corrélation pour le point $m2 = [108 ; 245]$. $w = N = 25$.