SIFT / Bag of Words

SIHAMDI Mostefa, BOUSBA Abdellah

UE RDFIA 2021-2022, Encadrants : Arthur Douillard, Alexandre Rame, Asya Grechka

M2 DAC

I. SIFT

- Question 1 :

Les deux masques de Sobel sont séparables $Mx = h_y.h_x^T$ et $Mx = h_x.h_y^T$ avec $hx = \frac{1}{2}(-1,0,1)$ et $hy = \frac{1}{2}(1,2,1)$

- Question 2 :

L'intérêt de séparer les masques est de réduire le cout de calcul, on travaille avec deux vecteurs de taille 3 au lieu d'une matrice 3 x 3.

- Question 3 :

L'utilisation du masque gaussien sert à réduire le bruit en supprimant les valeurs les plus éloignées du centre, cela permet de réduire les faux contours et de se concentrer sur les zones les plus importantes de chaque patch.

- Question 4 :

Le rôle de la discrétisation des directions de gradient est de faciliter la création de l'histogramme, aussi de rendre le descripteur plus robuste aux rotations.

- Question 5 :

La première étape qui consiste à annuler les descripteurs dont la norme euclidienne est inférieure à 0.5 permet de supprimer les descripteurs qui n'ont pas assez de contraste et éviter des informations non pertinentes. La normalisation et le seuillage à 0.2 rends le descripteur invariant au changement de luminosité.

- Question 6 :

Le modèle SIFT est robuste aux rotations, changements de luminosité et d'autres transformations géométriques ce qui nous permet de comparer des images prises dans des conditions différentes

- Question 7 :

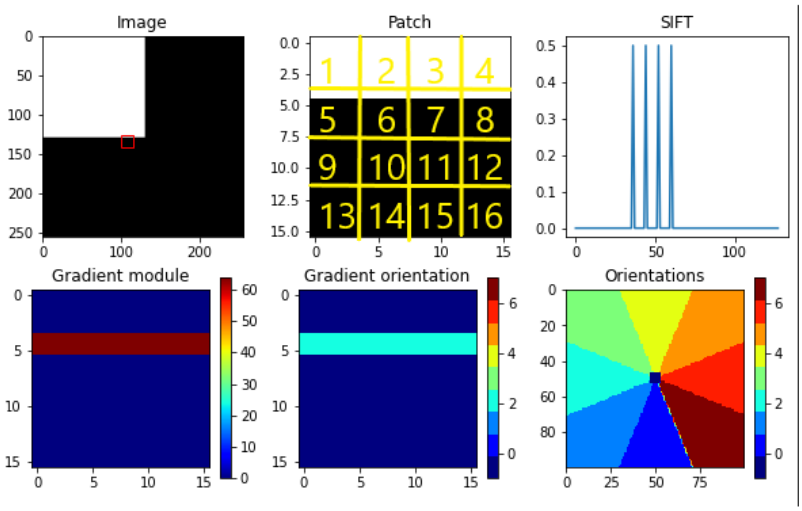


Figure 1 - Exemple SIFT

Le Patch de l'image ci-dessus représente une partie de l'image traité par SIFT, chaque Patch est découpé en 16 (4 x 4) régions. Les régions de 1-4 (resp. 9-16) sont entièrement blanches (resp. noires) donc leurs gradient est nul comme le montre la figure. En revanche, on remarque les 4 signaux des régions de 5-8 qui se trouve dans la zone de bordure entre les deux couleurs.

II. Visual Dictionary:

- Question 8 :

Le dictionnaire nous permet de réduire l'espace de représentations en rapprochant les vecteurs aux features, et donc nous pouvons comparez les images entre elles.

- Question 9 :

Comme la fonction est convexe il suffit d'annuler le gradient pour résoudre le problème d'optimisation $\min_c \sum_i \|x_i - c\|_2^2$.

$$grad = -2\sum_i (x_i - c) = 0$$

$$-2\sum_i x_i + 2\sum_i c = 0$$

$$-2\sum_i x_i + 2\,n.c = 0$$

$$c = \frac{1}{n}\sum_i x_i$$

- Question 10 :

Pour déterminer le k idéal nous pouvons soit utiliser la méthode « Elbow » qui réalise un grid search en prenant le ratio entre les distances intra-cluster et inter-cluster comme métrique, soit « Silhouette » qui prend la moyenne des distances comme métrique.

- Question 11 :

L'utilisation des SIFT au lieu des pixels directement est plus intéressante car les vecteurs SIFT sont plus informatifs. En d'autres termes après les traitements réalisés sur l'image le bruit et les régions non-pertinentes sont déjà supprimées et que les informations importantes sont retenues.

- Question 12 :

Les 3 images ci-dessous représentent chacune 16 régions plus proche des mots du dictionnaire choisi aléatoirement, pour chaque patch les régions sont similaires entre eux. Par exemple le premier patch représente ceux qui ont un bloc noir à gauche, la deuxième des lignes verticales et la troisième une texture qui se ressemble à l'herbe.

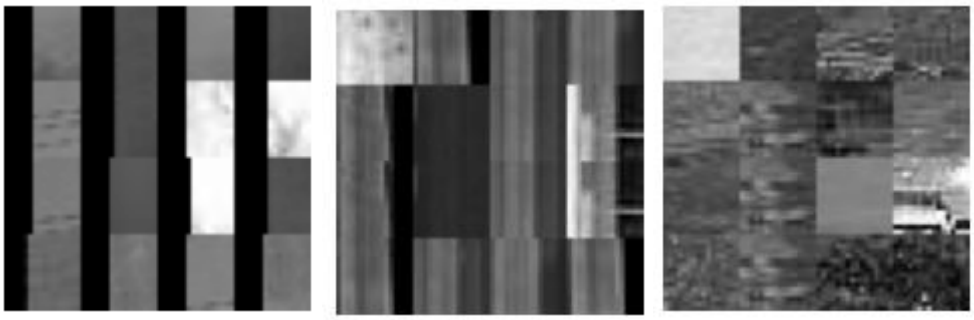


Figure 2 – exemple des régions proches des centres de cluster

III. Bag of Words (BoW) :

- Question 13 :

Le vecteur z est une représentation globale de notre image qui correspond au nombre des patch qui sont les plus proches du centre de chaque cluster

- Question 14 :

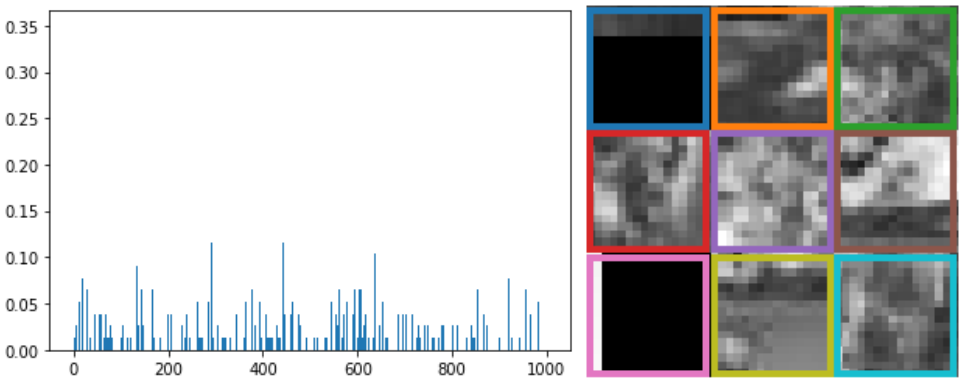


Figure 3 - Représentation du vecteur z

On remarque que plusieurs régions ont été bien classé comme similaire à un seul SIFT par exemple l'orange qui représente les feuilles de l'arbre ou le rouge qui représente l'herbe. Nous pouvons remarquer une hausse des valeurs dans l'histogramme pour ces régions ce qui est logique car ils sont très présents dans l'image.

- Question 15 :

L'avantage du codage au plus proche voisin est d'avoir une représentation légère et moins couteuse. Elle est aussi robuste à plusieurs transformations. Nous pouvons aussi utiliser un « soft-assignement » en donnant les probabilités d'appartenance de patch à chaque cluster au lieu de faire un one-hot encoding. Cela nous permettra d'améliorer l'expressivité et la stabilité.

- Question 16 :

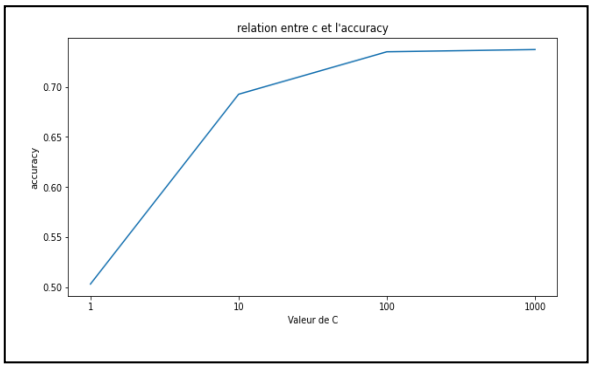
L'intérêt du « sum pooling » est de retourner le nombre de fois un cluster est apparu comme étant plus proche en d'autres termes favoriser les mots visuels qui apparaissent le plus. Nous pouvons aussi utiliser le « tf-idf » qui consiste à calculer non seulement la fréquence du mot mais aussi la fréquence inverse, ce qui est plus informatif comme un poids plus important sera donner aux mots moins fréquents.

- Question 17 :

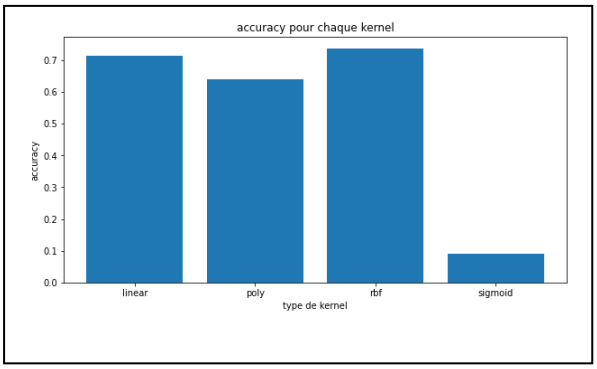
La normalisation L2 nous permet de comparer les vecteurs entre eux, elle ne perd pas la direction donc elle est invariante aux rotations. Cependant, ce n'est pas le cas pour la norme L1.

IV. Learning SVM classifier :

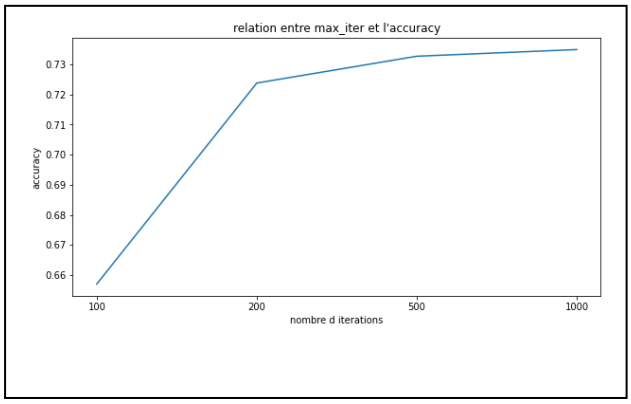
- Question 18 :



Le graph représente la valeur de C en fonction de l'accuracy , On remarque qu'à chaque fois le C augment l'accuracy augmente, dans notre cas on a pris les valeurs de C dans l'intervalle [1,1000],on peut trouver qu'avec des valeurs petit de C l'accuracy est très faible



Le graphe représente l'accuracy en fonction de type de karnel, on remarque que le karnel optimal est rbf qui donne une accuracy de 0.72,le karnel linaire donne des résultats proche de rbf avec une accuracy qui égal à 0.715,par contre le sigmoïde donne des faibles résultats.



le graphe représente l'accuracy en fonction de nombre d'itération ,En utilisant la stratégie OneVsall et en utilisant le karnel optimal (rbf) et le C optimal, On remarque que l'accuracy converge vers la valeurs de 0.735 tant que on augmente le nombre d'itération.

on fait un grid search sur tous les paramètres de notre svm (on prend rbf comme karnel par défaut puisqu'il donne des meilleurs performance).le tableau suivant représente un échantillon des résultats trouvé.

	kernel	C	max_iter	shrinking	probability	gamma	coef0	degree	accuracy
0	rbf	1	200	True	True	auto	/	/	0.56
1	rbf	1	200	True	True	scale	/	/	0.54
2	rbf	1	200	True	True	0.001	/	/	0.56
3	rbf	1	200	True	True	0.1	/	/	0.37
4	rbf	1	200	True	True	1.0	/	/	0.07
80	rbf	10	500	True	True	auto	/	/	0.59
81	rbf	10	500	True	True	scale	/	/	0.7
82	rbf	10	500	True	True	0.001	/	/	0.59
83	rbf	10	500	True	True	0.1	/	/	0.09
84	rbf	10	500	True	True	1.0	/	/	0.09
137	rbf	100	200	False	False	0.001	/	/	0.57
138	rbf	100	200	False	False	0.1	/	/	0.25
139	rbf	100	200	False	False	1.0	/	/	0.07
140	rbf	100	500	True	True	auto	/	/	0.59
141	rbf	100	500	True	True	scale	/	/	0.72

196	rbf	1000	200	False	False	scale	/	/	0.71
197	rbf	1000	200	False	False	0.001	/	/	0.57
198	rbf	1000	200	False	False	0.1	/	/	0.25
199	rbf	1000	200	False	False	1.0	/	/	0.07

On remarque que les paramètres qui donne des meilleurs résultats sont :
kernel=rbf, C=100, shrinking=False, probability= TRUE, gamma =scale
avec une accuracy qui égale à 0.72

- Question 19 :

L'effet de chaque paramètres :

C : Le paramètre C ajoute une pénalité pour chaque point de données mal classé. Si c'est petit, la pénalité pour les points mal classés est faible et une limite de décision avec une grande marge est choisie au détriment d'un plus grand nombre de mauvaises classifications. Si c'est grand, le SVM essaie de minimiser le nombre d'exemples mal classés en raison de la pénalité élevée, ce qui entraîne une limite de décision avec une marge plus petite.

Kernel : La fonction kernel est une sorte de mesure de similarité. Les entrées sont les caractéristiques originales et la sortie est une mesure de similarité dans le nouvel espace de fonctions. La similarité signifie ici un degré de proximité.

Gamma : Gamma est un paramètre de la fonction RBF qui contrôle la distance d'influence d'un seul point de training.

shrinking: shrinikng est utilisé pour accélérer l'optimisation. mais ils aident parfois, et parfois non.

- Question 20 :

Le dataset de validation est nécessaire pour optimise et ajuster les hyperparamètres de notre modèle, ces données doivent être distinctes avec les données de train et de test pour assurer une bonne optimisation et éviter le sur-apprentissages.