

Appariement de points par triangulation de Delaunay et réseau de neurones

Ratsimbazafy T.H.¹, Randriamitantsoa P.A.²

Laboratoire de recherche Télécommunication, Automatique, Signal Et Images (LR-T.A.S.I)

Ecole Doctorale en Sciences et Techniques de l'Ingénierie et de l'Innovation (ED -STII)

Équipe d'Accueil Doctorale Télécommunication, Automatique, Signal et Images (EAD-TASI)

BP 1500, Ankatso- Antananarivo 101 –Madagascar

¹tsiorha@yahoo.fr, ²rpauguste@gmail.com

Résumé

A partir d'un nuage de points pour lesquels la classe d'appartenance est connue, comment classer un nouveau point pour lequel cette classe est inconnue.

L'appariement de points pour obtenir un couple, triplet, quadruple ont été toujours utilisés dans le domaine de traitement d'images. Nous allons voir dans cet article la combinaison de méthode de triangulation de Delaunay avec un plus proche voisin, le tout avec un apprentissage de réseau de neurone.

Mots clés: Réseaux de neurones, Triangulation de Delaunay, proches voisins,

Abstract

From a cloud of points for which the class of membership is known, how to classify a new point for which this class is unknown.

The matching of points to obtain a couple, triplet, quadruple have always been successful in the field of image processing. We will see in this article the combination of Delaunay's triangulation method with a nearest neighbor, all with neural network learning.

Keywords: neural network, Delaunay's triangulation, nearest neighbor

1. Réseaux de neurones

Il existe de nombreux algorithmes dédiés à l'extraction de caractéristiques des images. Beaucoup d'entre eux fonctionnent de la même manière qu'un spiropgraphe. C'est à dire fait le tour en se cognant contre les bords jusqu'à ce qu'il couvre, avec un peu de chance, chaque point de l'image. Ainsi, un algorithme se déplace autour d'une image et en extrait des informations intéressantes. Ce processus est appelé détection de caractéristiques.

1.1 Définitions

Un réseau artificiel de neurones est constitué d'un pool d'unités de traitement simples qui communiquent par envoi sur un grand nombre de connexions pondérées.

On peut distinguer un ensemble d'aspects majeurs d'un modèle distribué parallèle comportant:

- un ensemble d'unités de traitement (neurones, cellules) ;
- un état d'activation y_k sur chaque unité, qui est équivalent à la sortie de l'unité ;
- les connexions entre les unités. En règle générale, chaque raccordement est renforcé par un poids w_{jk} qui détermine l'effet que le signal de l'unité j a sur l'unité k ;

- une règle de propagation, qui détermine l'entrée effective s_k d'une unité à partir de son entrée externe intrants ;
- une fonction d'activation F_k , qui détermine le nouveau niveau d'activation sur la base de la fonction l'entrée effective $s_k(t)$ et l'activation courante $y_k(t)$ (la mise à jour) ;
- une entrée externe (bias, offset) θ_k pour chaque unité ;
- une méthode de collecte de l'information (la règle d'apprentissage) ;
- un environnement dans lequel le système doit fonctionner, en fournissant des signaux d'entrée et, le cas échéant les signaux d'erreur nécessaires.

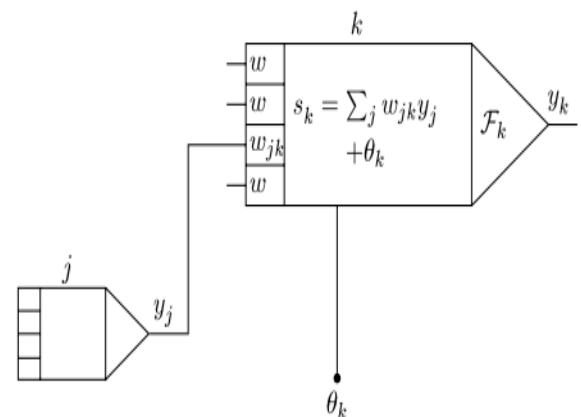


Figure 01 : Voisinage de pixel

1.2 Unité de traitement

Chaque unité accomplit un travail relativement simple : recevoir des commentaires de voisins ou de sources externes et l'utiliser pour calculer un signal de sortie qui se propage à d'autres unités. En dehors de cela une deuxième tâche est le réglage des poids. Le système est intrinsèquement parallèle en le sentiment que plusieurs unités peuvent effectuer leurs calculs en même temps.

Dans les systèmes neuronaux, il est utile de distinguer trois types d'unités : les unités d'entrée (indiquées par un index i) qui reçoivent des données provenant de l'extérieur du réseau neuronal, des unités de sortie (indiquées par un index o) qui envoient des données hors du réseau neuronal, et des unités cachées (indiquées par un index h) dont les signaux d'entrée et de sortie restent dans le réseau neuronal.

1.2.1 Connexions entre les unités

Dans la plupart des cas, nous supposons que chaque unité apporte une contribution supplémentaire à l'entrée de l'unité à laquelle elle est connectée.

L'entrée totale à l'unité k est simplement la somme pondérée des sorties séparées de

chacune des unités connectées plus un terme de polarisation ou de décalage θ_k

$$s_k(t) = \sum_j w_{jk}(t)y_j(t) + \theta_k(t) \quad (01)$$

La contribution pour w_{jk} positif est considérée comme une excitation et pour w_{jk} négatif comme une inhibition.

Dans certains cas, on utilise des règles plus complexes pour combiner les intrants, dans lesquelles une distinction est faite entre les intrants excitateurs et inhibiteurs. Nous appelons unités avec une règle de propagation unités sigma.

Une règle de propagation différente, introduite par Feldman et Ballard, est connue comme règle de propagation pour l'unité sigma-pi :

$$s_k(t) = \sum_j w_{jk}(t) \prod_m y_{jm}(t) + \theta_k(t) \quad (02)$$

Souvent, les y_{jm} sont pondérés avant multiplication. Bien que ces unités ne soient pas fréquemment utilisées, elles ont leur valeur pour l'entrée de données et l'implémentation de tables de consultation.

1.2.2 Règles d'activation et de sortie

Nous avons aussi besoin d'une règle qui donne l'effet de l'entrée totale sur l'activation de l'unité. Nous avons besoin d'une fonction F_k qui prend l'entrée totale $s_k(t)$ et l'activation courante $y_k(t)$ et produit une nouvelle valeur de l'activation de l'unité k :

$$y_k(t + 1) = F_k(y_k(t), s_k(t)) \quad (03)$$

Souvent, la fonction d'activation est une fonction non décroissante de l'entrée totale de l'appareil :

$$y_k(t + 1) = F_k(s_k(t)) \quad (04)$$

$$= F_k\left(\sum_j w_{jk}(t)y_j(t) + \theta_k(t)\right)$$

bien que les fonctions d'activation ne soient pas limitées à des fonctions non décroissantes.

Généralement, on utilise une fonction de seuil : une fonction de seuil de limitation dure (fonction sgn), ou une fonction linéaire ou semi-linéaire, ou un seuil de limitation progressive.

Pour cette fonction de limitation en douceur, souvent une fonction sigmoïde (en forme de S) comme

$$y_k = F(s_k) = \frac{1}{1 + e^{-s_k}} \quad (05)$$

est utilisé. Dans certaines applications, une tangente hyperbolique est utilisée, donnant des valeurs de sortie dans la plage $[-1 ; +1]$.

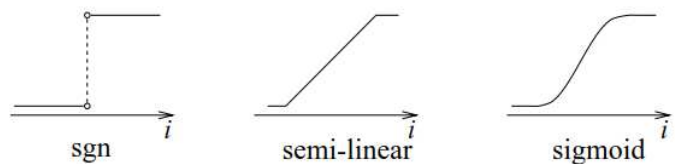


Figure 02 : Diverses fonctions d'activation pour une unité.

Dans certains cas, la sortie d'une unité peut être une fonction stochastique de l'entrée totale de l'unité. Dans ce cas, l'activation n'est pas déterminée de façon déterministe, mais par l'entrée neuronale détermine la probabilité p qu'un neurone obtienne une valeur d'activation élevée:

$$p(y_k \leftarrow 1) = \frac{1}{1 + e^{-s_k/T}} \quad (06)$$

dans laquelle T (température) est un paramètre qui détermine la pente de la fonction de probabilité.

Dans tous les réseaux que nous décrivons, nous considérons que la sortie d'un neurone est identique à son niveau d'activation.

1.3 Entraînement des réseaux de neurones artificiels

Un réseau neuronal doit être conçu de telle sorte que l'application d'un ensemble d'entrées produise (soit directement, soit par un processus de relâchement) l'ensemble souhaité de sorties.

Il existe différentes méthodes pour définir les points forts des connexions. L'une d'entre elles consiste à fixer les pondérations de façon explicite, à l'aide de connaissances a priori. Une autre façon est de " former " le réseau neuronal en l'alimentant de modèles d'enseignement et en lui permettant de changer son poids selon une règle d'apprentissage.

Nous pouvons classer les situations d'apprentissage en deux catégories distinctes. Celles-ci le sont:

- Apprentissage supervisé ou apprentissage associatif dans lequel le réseau est formé en lui fournissant des

modèles d'entrée et de sortie correspondants. Ces paires d'entrées-sorties peuvent être fournies par un enseignant externe ou par le système qui contient le réseau (auto-surveillé).

- Apprentissage non supervisé ou auto-organisationnel dans lequel une unité (de sortie) est formée pour répondre à des groupes de modèles à l'intérieur de l'entrée. Dans ce paradigme, le système est censé découvrir les caractéristiques statistiquement saillantes de la population d'entrée. Contrairement à l'apprentissage supervisé il n'existe pas d'ensemble a priori de catégories dans lesquelles les modèles doivent être classés ; le système doit plutôt développer sa propre représentation des stimuli d'entrée.

2. Triangulation de Delaunay

Delaunay était un mathématicien russe qui a étendu les travaux de Voronoï. La triangulation de Delaunay est un type de partitionnement d'un ensemble de points E positionnés dans un plan formé de triangles dont les sommets sont des objets, et qui à eux tous

constituent une partition de l'enveloppe convexe de ces objets.

La triangulation de Delaunay a deux propriétés principales :

- Le "critère du cercle": un triangle de Delaunay est un triangle qui a comme sommet trois objets, et tel que son cercle circonscrit n'ait en son intérieur aucun autre objet.
- La triangulation de Delaunay est parmi toutes les triangulations de E celle qui maximise l'angle minimum de tous les triangles.

Du point de vue de la théorie des graphes, la triangulation de Delaunay est définie comme le diagramme dual du diagramme de Voronoï de telle façon que deux objets sont liés par un arc dans la triangulation de Delaunay s'ils appartiennent à des régions de Voronoï adjacentes.

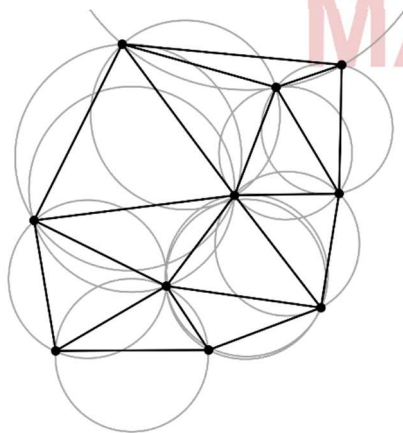


Figure 03 : *Delaunay triangulation*

Cette structure est très utile, par exemple, en mécanique, car c'est celle qui permet de mailler les objets de la manière la plus efficace, en minimisant les aires au carré des triangles. Elle est très utilisée aussi dans beaucoup d'autres domaines, dont celui qui nous occupe ici: le traitement d'image.

Tous les algorithmes de construction d'une triangulation de Delaunay reposent sur des opérations rapides permettant de déterminer la position d'un point par rapport à un cercle circonscrit à un triangle, et de structures de données efficaces pour conserver les triangles et les sommets.

Dans le plan, pour vérifier si un point D se trouve dans le cercle circonscrit à A, B et C, il suffit d'évaluer le déterminant de la matrice suivante :

$$\begin{vmatrix}
 A_x & A_y & A_x^2 + A_y^2 & 1 \\
 B_x & B_y & B_x^2 + B_y^2 & 1 \\
 C_x & C_y & C_x^2 + C_y^2 & 1 \\
 D_x & D_y & D_x^2 + D_y^2 & 1
 \end{vmatrix}
 \begin{matrix}
 (A_x^2 - D_x^2) + (A_y^2 - D_y^2) \\
 (B_x^2 - D_x^2) + (B_y^2 - D_y^2) \\
 (C_x^2 - D_x^2) + (C_y^2 - D_y^2)
 \end{matrix}
 > 0
 \tag{07}$$

Si A, B et C sont placés dans le sens antihoraire, le déterminant est positif si et seulement si D se trouve dans le cercle circonscrit.

En résumé, SURF ajoute de nombreuses caractéristiques pour améliorer la vitesse à chaque étape.

3. Algorithmes des plus proches voisins

Dans ce cadre, on dispose d'une base de données d'apprentissage constituée de N couples « entrée-sortie ». Pour estimer la sortie associée à une nouvelle entrée x, la méthode des k plus proches voisins consiste à prendre en compte (de façon identique) les k échantillons d'apprentissage dont l'entrée est la plus proche de la nouvelle entrée x, selon une distance à définir. Puisque cet algorithme est basé sur la distance, la normalisation peut améliorer sa précision.

La méthode la plus simple consiste à associer à l'élément à classer, le label de l'élément le plus proche dans l'ensemble Ceci mène à l'algorithme de classification suivant :

Soit

$$X = (x_1 \dots x_N) \subset E \tag{08}$$

Un ensemble d'éléments d'un espace métrique quelconque,

Soit $(c(x_1), \dots, c(x_N))$ les classes associées à chacun des éléments de X. On note d la distance définie sur l'espace métrique E. Soit x un élément à classer, on cherche à déterminer la classe $\hat{c}(x)$ associée à x. On définit comme étant :

$$x_{i^*} = \arg \min_{i \in \{1, \dots, N\}} d(x_i, x) \tag{09}$$

Alors $\hat{c}(x) = c(x_{i^*})$

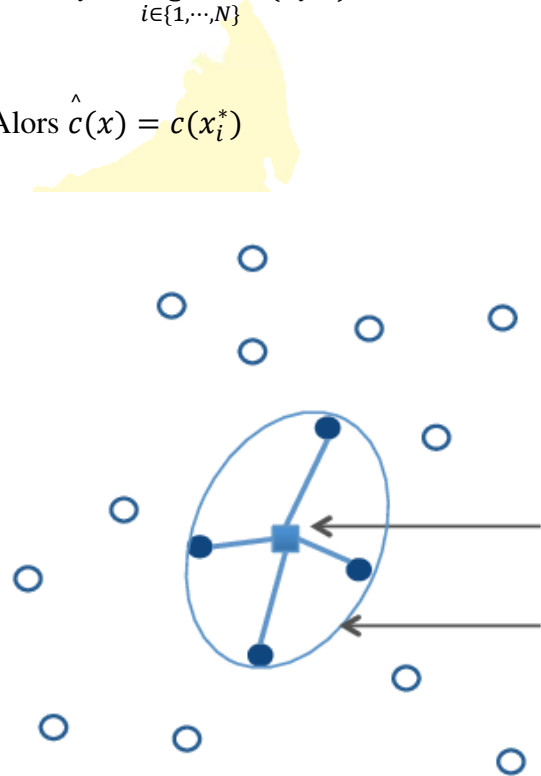


Figure 04 : Illustration KNN avec k=4

Dans le cas de l'ouvrage nous allons faire l'appariement deux à deux entre les points les plus proches .

4. Simulations

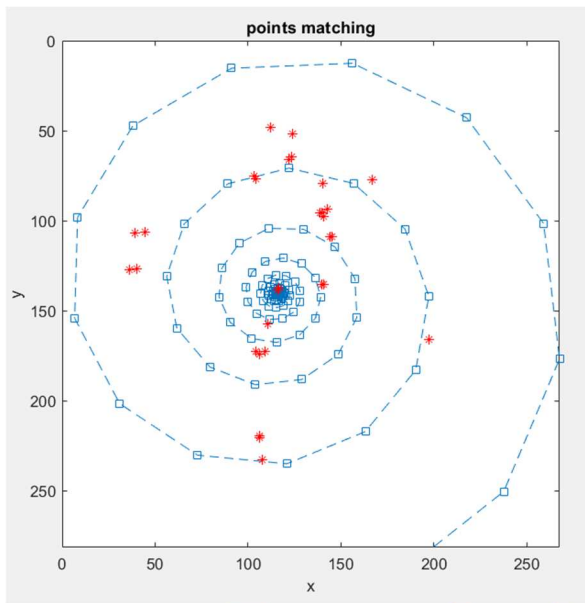


Figure 05 : Images test

Dans ce premier temps, nous avons les deux classes de points : en rouge et en bleu

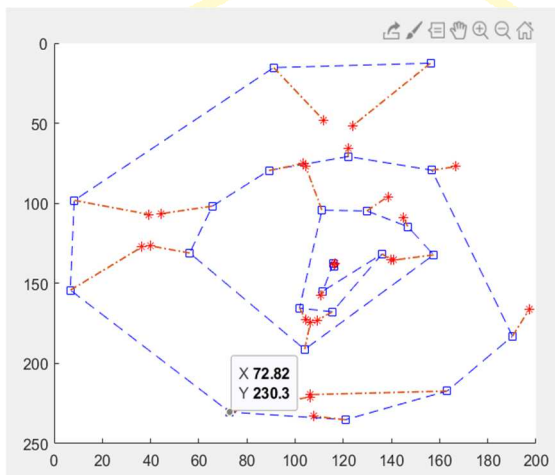


Figure 06 : Les couples de points retenues

Dans un second temps, l'algorithme va faire le test des points rouges qui se trouvent être le plus proche des points bleus.

2.1 Résultats et interprétation La mesure des résultats de correspondances utilisée est le temps : nous allons voir la performance de l'algorithme sans réseau de neurone et avec réseau de neurone lors de l'appariement.

Voici les figures du temps résumant les tests :

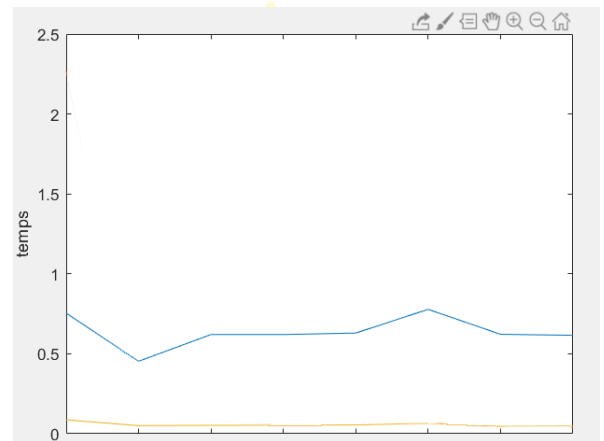


Figure 07 : Aperçu de la différence de temps

Nous pouvons voir en bleu la courbe de temps de l'algorithme de triangulation combiné avec le plus proche voisin sans utilisation de réseaux de neurones.

Nous pouvons voir en orange la courbe de temps de l'algorithme de triangulation combiné avec le plus proche voisin avec utilisation de réseaux de neurones.

Le réseau de neurones a été entraîné avec 70 itérations pour aboutir à ce résultat.

5. Conclusion

Le but de tout système utilisant des calculs mathématiques est de raccourcir le temps de son exécution. L'efficacité du système de triangulation de Delaunay et du plus proche voisin a été prouvée dans cet ouvrage. Mais en ajoutant un réseau de neurone au système nous pouvons obtenir un gain de temps beaucoup plus grand. Le système sera ainsi plus performant. L'inconvénient du réseau de neurone est le temps d'apprentissage durant lequel il est entraîné mais une fois le modèle acquis ce temps est largement réduit.

BIBLIOGRAPHIE

- [1] R.Venkatesan,B.Li, «*Convolutional Neural Networks in Visual Computing: A Concise Guide*», CRC Press, 2018.
- [2] M.Yedroudj, M.Chaumont, F.Comby, «*Yedroudj-Net:un réseaux de neurones efficace pour la stéganalyse spatiale.*», HAL, 2018.
- [3] M.Berg,O.Cheong,M.v.Kreveld, M.Overmars,“*Computational Geometry: Algorithms and Applications*”, Berlin, Springer-Verlag, 2008.
- [4] N.P.Weatherill,«*Delaunay triangulation in computational fluid dynamics*», Computers & Mathematics with Applications, vol. 24,1992.
- [5] S.Piryonesi ,Tamer ,E. El-Diraby, « *Role of Data Analytics in Infrastructure Asset Management: Overcoming Data Size and Quality Problems* », Journal of Transportation Engineering, 2020.
- [6] Hastie,Trevor,Friedman,J.H.Jerome H.), “*The elements of statistical learning : data mining, inference, and prediction*”, Springer, 2001