

Combinaison d'ordres totaux stochastiques spatiaux-spectraux pour le traitement nonlocal morphologique

Olivier LÉZORAY

Normandie Univ, UNICAEN, ENSICAEN, CNRS, GREYC, 14000 Caen, France

olivier.lezoray@unicaen.fr

Résumé – L'extension de la morphologie mathématique aux données multivariées est un sujet de recherche actif ces dernières années. Dans cet article, nous proposons une approche qui repose sur la combinaison consensuelle de plusieurs ordres de permutation stochastiques. Ces derniers sont obtenus en recherchant un chemin le plus court possible sur un graphe représentant une image. La construction du graphe peut être basée sur des informations spatiale et spectrale et permet naturellement un traitement non local basé sur des patches.

Abstract – The extension of mathematical morphology to multivariate data has been an active research topic in recent years. In this paper we propose an approach that relies on the consensus combination of several stochastic permutation orderings. The latter are obtained by searching for a smooth shortest path on a graph representing an image. The construction of the graph can be based on both spatial and spectral information and naturally enables patch-based nonlocal processing.

1 Introduction

Les opérateurs de la morphologie mathématique (MM) reposent sur la notion de treillis complet (\mathcal{T}, \leq) [1] qui est un ensemble équipé d'une relation d'ordre total, de sorte que chaque sous-ensemble \mathcal{P} de \mathcal{T} a une limite inférieure $\wedge \mathcal{P}$ et une limite supérieure $\vee \mathcal{P}$. Dans ce contexte, les images sont modélisées par des fonctions $f : \Omega \subset \mathbb{Z}^l \rightarrow \mathcal{T} \subset \mathbb{R}^n$ où l est la dimension de l'image, n le nombre de canaux, et \mathcal{T} est un ensemble de vecteurs. Si la MM est bien définie pour les fonctions scalaires, il n'existe pas d'extension générale admise permettant de considérer des vecteurs puisqu'il n'existe pas d'ordre naturel des vecteurs [2]. Pour faire face ce problème, plusieurs approches ont été proposées [3, 4, 5] et les plus récentes définissent une relation d'ordre entre les vecteurs d'un ensemble \mathcal{T} par l'utilisation de h -ordres [6]. Ceci correspond à considérer une transformation h de \mathcal{T} vers \mathcal{L} , un treillis complet équipé de l'ordre lexicographique [6]. Soit \leq_h le h -ordre $h : \mathcal{T} \rightarrow \mathcal{L}$ défini par : $\forall (\mathbf{v}_i, \mathbf{v}_j) \in \mathcal{T} \times \mathcal{T}, \mathbf{v}_i \leq_h \mathbf{v}_j \Leftrightarrow h(\mathbf{v}_i) \leq h(\mathbf{v}_j)$. Alors, \mathcal{T} n'a plus besoin d'être un treillis complet, puisque l'ordre sur \mathcal{T} peut être induit sur \mathcal{L} au moyen de h [3]. Quand h est bijective, cela correspond à définir une courbe de remplissage d'espace [7]. Récemment, plusieurs travaux ont proposé de définir un h -ordre, non pas de manière générale, mais relativement à une image donnée [8]. Ainsi, certaines approches ont cherché à définir des ordres totaux qui sont adaptés uniquement à l'ensemble \mathcal{T} des vecteurs spectraux d'une image [4]. En effet, certains vecteurs de l'ensemble \mathcal{T} n'auront jamais à être comparés puisqu'ils n'apparaîtront pas simultanément dans un élément structurant. Par conséquent, il peut être intéressant d'avoir un ordre qui tienne compte de la proximité spatiale en plus de la proximité spectrale, comme abordé dans

[9]. Dans cet article, nous proposons un nouvel ordre spatio-spectral basé sur la combinaison de plusieurs ordres stochastiques (appelés aussi permutations) qui construisent chacun un chemin Hamiltonien sur une image. Le document est organisé comme suit. Dans la section 2 nous présentons la construction d'un chemin Hamiltonien sur une image avec un ordre de permutation 1D sur des graphes grille de pixels. Nous exposons ensuite dans la section 3 comment cet ordre peut prendre en compte la proximité spatiale et spectrale avec l'utilisation de graphes non locaux. Puisque l'ordre proposé dépend d'un pixel de départ, plusieurs ordres sont obtenus qui sont combinés en utilisant une approche consensuelle dans la section 4. Les dernières sections présentent les résultats et concluent.

2 Ordre de permutation

Nous considérons que le domaine Ω de l'image est un graphe grille $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ où les sommets $\mathcal{V} = \{v_1, \dots, v_m\}$ correspondent aux pixels et les arêtes $e_{ij} = (v_i, v_j)$ connectent les noeuds en 8-adjacence. La notation $v_i \sim v_j$ est utilisée pour désigner deux sommets adjacents. Avec cette définition, les images sont représentées sous forme de signaux sur graphes qui associent des vecteurs à des sommets et sont définis par $f : \mathcal{G} \rightarrow \mathcal{T} \subset \mathbb{R}^n$ où \mathcal{T} est un ensemble de vecteurs (nous ne considérerons que les vecteurs couleur RGB, c'est-à-dire $n = 3$). À chaque sommet $v_i \in \mathcal{G}$ est associé un vecteur $\mathbf{v}_i = f(v_i)$. L'ensemble $\mathcal{T} = \{\mathbf{v}_1, \dots, \mathbf{v}_m\}$ désigne les vecteurs associés à tous les sommets du graphe (avec une numérotation row-major des pixels de l'image). Nous utiliserons la notation $\mathcal{T}[i] = \mathbf{v}_i$ pour désigner le i -ème élément d'un ensemble. Pour construire un ordre h adapté à l'image, nous pro-

posons de construire une courbe de remplissage d'espace [7] sur le graphe grille \mathcal{G} . Cela correspond à la construction d'un chemin Hamiltonien sur toute l'image : un chemin qui passe par tous les sommets du graphe grille et qui ne traverse chaque sommet qu'une fois. Étant donné l'ensemble \mathcal{T} , la construction d'un chemin Hamiltonien revient à définir une permutation triée $\mathcal{P} = \mathbf{P}\mathcal{T}$ des vecteurs de \mathcal{T} avec \mathbf{P} une matrice de permutation de taille $m \times m$. Soit σ une permutation d'index $\mathcal{I} = \{1, \dots, m\}$. Si $\sigma(i) = j$, alors $\mathbf{P}_{ij} = 1$ et 0 sinon. La permutation induite est $\mathcal{P} = \mathbf{P}\mathcal{T} = \{\mathbf{v}_{\sigma^{-1}(1)}, \dots, \mathbf{v}_{\sigma^{-1}(m)}\}$ où $(\sigma^{-1} \circ \sigma)$ représente l'identité. Une permutation quelconque n'est pas intéressante, et des contraintes spatiales et spectrales doivent être prises en compte [8]. A cette fin, nous considérons la construction d'une permutation lisse. L'aspect lisse d'un ensemble ordonné s'exprime par la Variation Totale (TV) de ses éléments :

$$\|\mathcal{T}\|_{TV} = \sum_{i=1}^{m-1} \|\mathbf{v}_i - \mathbf{v}_{i+1}\| \quad (1)$$

L'opérateur de permutation optimal \mathbf{P} peut être obtenu en minimisant la variation totale de $\mathbf{P}\mathcal{T}$:

$$\mathbf{P}^* = \arg \min_{\mathbf{P}} \|\mathbf{P}\mathcal{T}\|_{TV} \quad (2)$$

Ce problème d'optimisation fournit une permutation telle que le chemin Hamiltonien correspondant est le plus court et est équivalent à résoudre le problème du voyageur de commerce. Ce problème étant trop difficile pour des grands ensembles, il peut être résolu en utilisant une version randomisée de l'heuristique des plus proches voisins présentée dans [10]. Cet algorithme de permutation part d'un sommet arbitraire et construit le chemin en choisissant un sommet au hasard parmi ses deux sommets voisins les plus proches inexplorés.

Après la construction de la permutation, nous définissons l'ordre h à partir de la permutation construite comme $h(\mathbf{v}_i) = \sigma(i)$ et ceci définit le treillis complet (\mathcal{T}, \leq_h) . Avec un signal sur graphe $f : \mathcal{G} \rightarrow \mathcal{T}$, une nouvelle représentation est obtenue sous la forme de la paire (I, \mathcal{P}) avec $I(v_i) = \sigma(i)$. Lorsqu'un signal sur graphe est codé de cette façon, l'information spectrale n'est pas directement portée par l'index I , mais est stockée dans une donnée séparée appelée palette : l'ensemble \mathcal{P} des vecteurs triés. La figure 1 présente la permutation et l'index obtenus à partir d'une image originale. On note que la permutation \mathcal{P}_i est présentée avec la même taille que l'index car elle contient toutes les couleurs de l'image originale. Un inconvénient de l'ordre décrit est sa nature gloutonne. En effet, comme on peut le voir dans la Figure 1, la dernière partie de la permutation n'est pas lisse.

Le signal sur graphe original f peut être directement récupéré puisque $f(v_i) = \mathcal{P}[I(v_i)] = \mathcal{T}[i] = \mathbf{v}_i$. Comme exposé dans [4], à partir d'une telle représentation, des opérations morphologiques peuvent être définies et appliquées au signal. L'érosion et la dilatation d'un signal f au sommet $v_i \in \mathcal{G}$ par un élément structurant $B_k \subset \mathcal{G}$ sont définies comme :

$$\epsilon_{B_k}(f)(v_i) = \{\mathcal{P}[\wedge I(v_j)], v_j \in B_k(v_i)\} \quad (3)$$

$$\delta_{B_k}(f)(v_i) = \{\mathcal{P}[\vee I(v_j)], v_j \in B_k(v_i)\} . \quad (4)$$

Un élément structurant $B_k(v_i)$ de taille k défini à un sommet v_i correspond à l'ensemble des sommets accessibles à partir de v_i en k sauts.

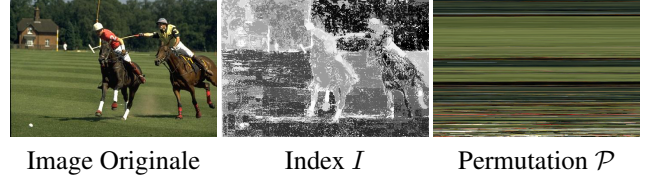


FIGURE 1 – Exemple d'ordre de permutation.

3 Ordres spatiaux-spectraux

Dans l'algorithme de permutation de [10], il n'y a aucune restriction sur le graphe considéré. Cependant, dans la section 2, nous nous sommes limités à des graphes grille en 8-connexité, mais il n'y a aucune raison de le faire. Dans cette section, nous considérons plusieurs graphes possibles pour la construction de la permutation. Le graphe ne peut utiliser que des informations spatiales ou spectrales, ou les deux :

- Graphe grille en 8-adjacence (noté \mathcal{G}_0) : cette construction relie chaque sommet à ses 8 plus proches voisins spatialement.
- Graphe en B -adjacence (noté \mathcal{G}_B) : cette construction relie chaque sommet v_i à tous les sommets contenus dans une fenêtre de taille $(2B + 1) \times (2B + 1)$ autour de v_i .
- Graphe des K -plus proches voisins (noté \mathcal{G}_K^s) : cette construction relie chaque sommet v_i à ses K plus proches voisins (en termes de distance spectrale) parmi l'ensemble des sommets.

Les deux premiers types de graphes \mathcal{G}_0 et \mathcal{G}_B ne considèrent que l'information de proximité spatiale pour relier les sommets alors que le graphe \mathcal{G}_K^s ne considère que l'information de proximité spectrale et est un graphe non local. Il peut être intéressant de construire des graphes qui combinent l'utilisation des proximités spatiale et spectrale pour inférer l'ensemble des arêtes. Pour le graphe \mathcal{G}_K^s , ceci est même essentiel car sa construction ne peut pas assurer que le graphe soit connecté et l'algorithme de permutation ne sera pas capable d'atteindre tous les vertices et se terminer. Ceci peut être facilement obtenu en considérant l'union de plusieurs ensembles d'arêtes issues des graphes précédemment considérés. Pour de telles constructions de graphe, avec l'algorithme de permutation, on obtient alors un ordre de permutation spectral-spatial. Cela peut être très intéressant car les proximités spatiales et spectrales sont prises en compte pour construire la permutation, ce qui peut réduire la complexité des calculs et assurer que les voisins les plus proches sont trouvés. En outre, la proximité spectrale utilisée pour déterminer les voisins les plus proches dans l'ensemble des sommets adjacents peut prendre en compte toute mesure de distance. En particulier, au lieu d'utiliser une distance classique L_2 entre les vecteurs \mathbf{v}_i associés aux sommets, on peut considérer des patches $\mathbf{p}_i^w = [f(x_i + t), \forall t \in [-w/2, w/2]^{2T}]^T$

de taille $w \times w$ pour obtenir un ordre plus lisse.

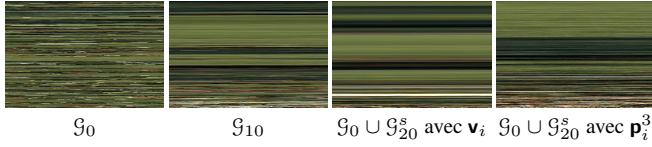


FIGURE 2 – Illustration de l’influence de la construction du graphe sur la permutation obtenue.

La figure 2 présente plusieurs ordres de permutation obtenus avec différents graphes. On peut voir qu’avec \mathcal{G}_0 la permutation n’est pas très lisse et le résultat obtenu avec un graphe plus connecté est bien meilleur (graphe \mathcal{G}_{10}). Enfin, l’ordre de permutation construit peut être très différent en considérant à la fois l’information spatiale et spectrale. Les dernières colonnes de la figure 2 illustrent ceci avec deux permutations construites sur l’union d’un graphie grille \mathcal{G}_0 et d’un graphe des plus proches voisins \mathcal{G}_{20}^s avec une mesure de similarité spectrale effectuée soit avec des vecteurs couleur \mathbf{v}_i ou bien des patches \mathbf{p}_i^3 de taille 3×3 .

4 Consensus d’ordres

L’algorithme de permutation est un processus aléatoire qui part d’un sommet arbitraire du graphe. Par conséquent, à chaque exécution de l’ordre de permutation, si un sommet de départ différent est choisi, un résultat différent est obtenu. Dans [10], les auteurs en ont tiré le meilleur parti en utilisant plusieurs permutations pour le débruitage. Nous proposons de procéder différemment et de combiner plusieurs ordres de permutation en un seul. Nous désignons par h_i une permutation donnée et nous considérons un ensemble de l permutations disponibles. Le principe consistant à combiner plusieurs ordres (également appelés rangs ou permutations) en un seul est connu sous le nom d’agrégation de rang, dont le but est de fournir un classement consensuel [11]. Trouver la permutation de consensus peut être formulé comme un problème d’optimisation, qui est généralement NP-dur [12] et des méthodes heuristiques ont été proposées. Dans cet article nous considérons les suivantes (voir [11] pour plus de détails sur celles-ci). L’agrégation Instant-Runoff détermine le rang de l’élément qui a le moins de votes en sa faveur et répète ceci sur les éléments restants. Le processus est le suivant. On commence par compter les votes de position en première place selon les ordres avec $R(\mathbf{v}_j) = \sum_{i=1}^l \delta_i(\mathbf{v}_j)$ où $\delta_i(\mathbf{v}_j)$ indique un vote de position en premier rang selon la permutation h_i et est défini par

$$\delta_i(\mathbf{v}_j) = \begin{cases} 1 & \text{si } \mathbf{v}_j \geq_{h_i} \mathbf{v}_k, \forall k \neq j \\ 0 & \text{sinon} \end{cases} \quad (5)$$

L’élément ayant obtenu le moins de votes se voit attribuer la position la plus basse dans le classement consensuel obtenu. Le Borda count détermine la position de classement de tous les éléments simultanément. Elle assigne à chaque élément \mathbf{v}_j ($j \in \mathcal{I}$) un score $B^i(\mathbf{v}_j)$ basé sur les positions fournies par les

permutations h_i ($i = 1, \dots, l$) : $B^i(\mathbf{v}_j) = 1 - \frac{h_i(\mathbf{v}_j)-1}{m}$. Selon les scores, le Borda count génère un score final en utilisant l’agrégation moyenne $BS(\mathbf{v}_j) = \frac{1}{l} \sum_{k=1}^l B^k(\mathbf{v}_j)$. Le classement fusionné est obtenu en triant les scores agrégés des items. Nous proposons de modifier le Borda count en Borda count pondéré afin de tenir compte de l’aspect lisse de chaque permutation. En effet, dans notre cas, nous agrégeons les ordres de permutation qui représentent les chemins lisses les plus courts et nous aimerions obtenir une permutation consensuelle qui soit également lisse. Pour ce faire, nous considérons un score pondéré défini par $B_s^i(\mathbf{v}_j) = B^i(\mathbf{v}_j) \times \nabla \mathcal{P}_i(\mathbf{v}_j)$ où $\nabla \mathcal{P}_i(\mathbf{v}_j)$ est un gradient 1D centré mesuré avec les voisins droite et gauche de \mathbf{v}_j dans la permutation \mathcal{P}_i . Un gradient fort caractérisant un aspect non lisse, cet élément recevra score plus haut afin de le décaler vers la fin de l’ordre. La figure 3 illustre le principe du consensus d’ordres. Étant donné une image originale, cinq ordres de permutation différents ($\mathcal{P}_1, \dots, \mathcal{P}_5$) ont été générés en utilisant différents noeuds de départ. Ils sont montrés dans les premières lignes de la Figure 3. Les permutations obtenues peuvent être très différentes les unes des autres en raison de la nature stochastique de l’algorithme de construction de la permutation. La dernière ligne de la Figure 3 présente les différentes permutations de consensus qui peuvent être obtenues avec Instant-Runoff, Borda count ou Borda count pondéré. Comme on peut le constater, le Borda count permet d’obtenir un consensus plus harmonieux que l’Instant-Runoff, ce qui est renforcé par la version proposée du Borda count pondéré.

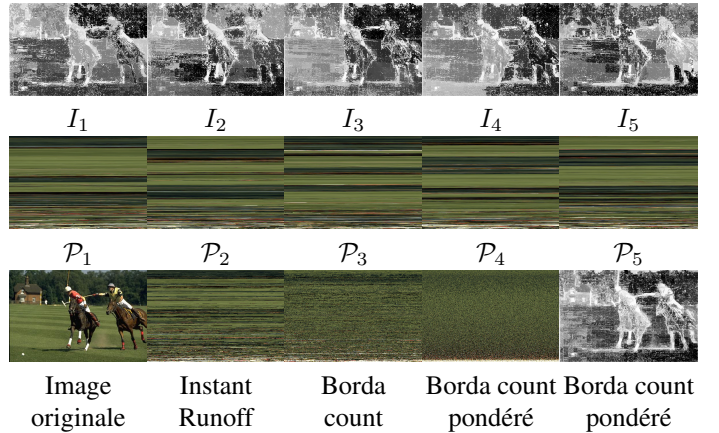


FIGURE 3 – Combinaison par consensus de plusieurs ordres stochastiques.

5 Résultats

Maintenant que nous disposons d’une méthode permettant de générer une permutation de consensus à partir de plusieurs permutations stochastiques, nous sommes en mesure d’utiliser la permutation obtenue pour effectuer un traitement morphologique des images. Dans cette section, nous étudions l’influence des différents paramètres de l’approche : la méthode de com-

binasion, le graphe construit, la mesure de proximité spectrale. La figure 4 présente une illustration de l'influence de ces paramètres avec le résultat d'une opération de fermeture $\phi = \epsilon\delta$ avec un élément structurant carré de taille 5×5 . Le nombre de permutations combinées est de 5 et le graphe considéré est \mathcal{G}_{10} .

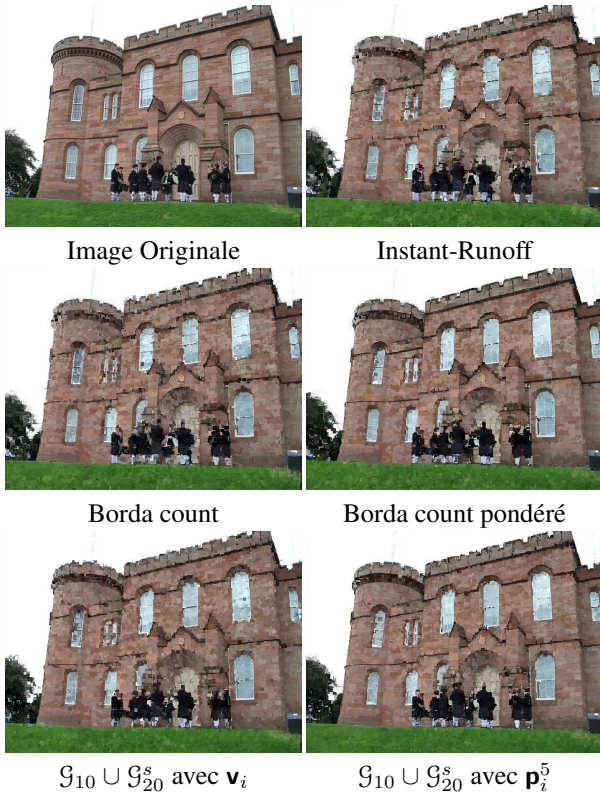


FIGURE 4 – Influence des paramètres sur une fermeture morphologique.

Comme le montrent les deux premières lignes de la Figure 4, il y a peu de différences entre les combinaisons consensuelles Instant-Runoff et Borda, et les fermetures avec ces méthodes de combinaison ne sont pas capables de bien préserver les bords des objets dans l'image. Ce n'est pas le cas pour le Borda count pondéré où les bords sont préservés et bien délimités, comme attendu pour un traitement morphologique. La dernière ligne de la Figure 4 présente l'influence de la construction du graphe. Ceci correspond à la combinaison avec le Borda count pondéré de 5 permutations stochastiques spatio-spectrales avec l'utilisation de vecteurs de couleurs ou de patches de taille 5×5 pour la mesure de proximité spectrale. Comme on peut le voir, la combinaison spatiale et spectrale peut être intéressante, mais seulement lorsque la mesure de proximité est suffisamment précise. En effet, avec les vecteurs de couleur, le résultat obtenu ne diffère pas beaucoup de celui avec seulement des contraintes spatiales pour la construction du graphe. Inversement, l'utilisation de patches a permis d'améliorer le résultat, beaucoup plus contrasté, et de mieux préserver les bords et la texture. Ceci offre une nouvelle alternative pour le traitement morphologique non local des images.

6 Conclusion

Cet article a introduit une nouvelle approche pour le traitement morphologique non local des images. Plusieurs ordres de permutation stochastiques sont d'abord construits de manière à obtenir des chemins lisses sur un graphe. La construction du graphe peut bénéficier à la fois d'informations spatiales et spectrales et permet d'obtenir un ordre de permutation qui repose sur des patches au lieu de couleurs. Les ordres de permutation sont combinés en un seul en utilisant une approche Borda count pondérée pour produire un ordre de permutation consensuel lisse.

Références

- [1] C. Ronse, "Why mathematical morphology needs complete lattices," *Signal Processing*, vol. 21, no. 2, pp. 129–154, 1990.
- [2] V. Barnett, "The ordering of multivariate data," *J R STAT SOC SER A-G*, vol. 139, no. 3, pp. 318–355, 1976.
- [3] E. Aptoula and S. Lefèvre, "Multivariate mathematical morphology applied to colour image analysis," in *Multivariate image processing : methods and applications*, C. Collet, J. Chanussot, and K. Chehdi, Eds. ISTE - John Wiley, 2009, pp. 303–337.
- [4] O. Lézoray, "Complete lattice learning for multivariate mathematical morphology," *J VIS COMMUN IMAGE R*, vol. 35, pp. 220–235, 2016.
- [5] S. Velasco-Forero and J. Angulo, "Mathematical morphology for vector images using statistical depth," in *ISMM*, vol. LNCS 6671, 2011, pp. 355–366.
- [6] J. Goutsias, H. Heijmans, and K. Sivakumar, "Morphological operators for image sequences," *COMPUT VIS IMAGE UND*, vol. 62, no. 3, pp. 326–346, 1995.
- [7] J. Chanussot and P. Lambert, "Bit mixing paradigm for multivalued morphological filters," in *International Conference on Image Processing and Its Applications*, vol. 2, 1997, pp. 804 – 808.
- [8] E. Chevallier and J. Angulo, "Image adapted total ordering for mathematical morphology on multivariate images," in *IEEE ICIP*, 2014, pp. 2943–2947.
- [9] M. A. Veganzones, M. D. Mura, G. Tochon, and J. Chanussot, "Binary partition trees-based spectral-spatial permutation ordering," in *ISMM*, 2015, pp. 434–445.
- [10] I. Ram, M. Elad, and I. Cohen, "Image processing using smooth ordering of its patches," *IEEE T IMAGE PROCESS*, vol. 22, no. 7, pp. 2764–2774, 2013.
- [11] J. Ding, D. Han, J. Dezert, and Y. Yang, "A new hierarchical ranking aggregation method," in *FUSION*, 2016, pp. 1562–1569.
- [12] C. Dwork, R. Kumar, M. Naor, and D. Sivakumar, "Rank aggregation methods for the web," in *WWW*. ACM, 2001, pp. 613–622.