HAL
open science

Traitement non-local de nuages de points colorés 3D : opportunités pour l' héritage culturel

François Lozes, Abderrahim Elmoataz, Olivier Lézoray

To cite this version:

François Lozes, Abderrahim Elmoataz, Olivier Lézoray. Traitement non-local de nuages de points colorés 3D : opportunités pour l' héritage culturel. GRETSI, 2015, Bordeaux, France. hal-01252876

HAL Id: hal-01252876

https://hal.science/hal-01252876v1

Submitted on 9 Jan 2016

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Traitement non-local de nuages de points colorés 3D : opportunités pour l'héritage culturel.

François LOZES, Abderrahim ELMOATAZ, Olivier LÉZORAY

Normandie Univ., UNICAEN, ENSICAEN, GREYC UMR CNRS 6072, Caen
{francois.lozes,abderrahim.elmoataz-billah,olivier.lezoray}@unicaen.fr

Résumé – Avec les récentes avancées technologiques 3D, la communauté du patrimoine culturel s'intéresse de plus en plus à la numérisation d'objets du patrimoine tels que les objets antiques, des artéfacts ou des sites archéologiques du patrimoine. Un des objectifs du patrimoine culturel, concerne la préservation et l'analyse d'objets anciens. Nous proposons une méthode simple de traitement et d'analyse à base d'EDPs (Équations aux Dérivées Partielles) sur nuages de points 3D en utilisant le cadre des EdPs (Équations aux différences Partielles) sur graphes. Puis, nous présentons des exemples d'applications sur des nuages de points colorés d'objets réels du patrimoine culturel 3D.

Abstract – With the advance of 3D scanning technology, the cultural heritage community is increasingly interested in 3D scans of cultural objects such as antiques, artifacts and heritage sites. Digitization of these objects is commonly aimed at heritage preservation. Our contributions are twofold. First, we propose a simple method to solve PDEs (Partial Differential Equations) on point clouds using the framework of PdEs (Partial difference Equations) on graphs. Second, we survey several applications of 3D color point cloud processing on real examples.

1 Introduction

Avec les récentes avancées technologiques 3D, la numérisation d'objets 3D est devenue une pratique courante dans beaucoup de domaines, en particulier dans le domaine du patrimoine culturel. En effet, les nuages de points obtenus par les scanneurs 3D permettent de représenter des objets de différentes tailles, de différentes géométries et à des précisions parfois très élevées.

Contrairement aux images et aux vidéos, les nuages de points 3D ne sont pas organisés sur une grille Cartésienne. Les outils de traitement de signal traditionnels ne peuvent donc pas être utilisés directement sur un nuage de points colorés 3D, car aucune information sur la géométrie n'est disponible [AJW13]. De plus, il n'est pas désirable ne transformer le nuage de points 3D en un maillage, car cela engendrerait une perte de précision qui est incompatible avec l'objectif de préservation de l'objet.

Nous proposons une approche simple pour adapter et résoudre des EDPs sur nuage de points 3D. Cette méthode repose sur le cadre des EdPs sur des graphes pondérés. Ainsi, des problèmes inverses sur nuages de points, tels que la restauration ou l'interpolation, peuvent être formulées et résolues directement sur graphes. Nous présenterons plusieurs applications sur des nuages de points colorés d'objets 3D réels du patrimoine culturel.

2 Construction de graphes à partir d'un nuage de points

L'approche proposée pour traiter des nuages de points, commence par créer un graphe pondéré à partir d'un nuage de point donné. Cette section montre comme construire un tel graphe : d'abord, la topologie du graphe est définie, ensuite les arêtes du graphe sont pondérées.

2.1 Construction du graphe

Considérons un nuage de points P comme un ensemble de vecteurs $\{\mathbf{p}(v_1), \ldots, \mathbf{p}(v_n)\} \in \mathbb{R}^3$. Chaque point $p \in P$, est associé à un sommet d'un graphe $\mathcal{G}$ afin de définir un ensemble de sommets $\mathcal{V} = \{v_1, v_2, \ldots, v_n\}$. Ensuite, pour déterminer l'ensemble des arêtes $\mathcal{E}$ du graphe $\mathcal{G}$, nous définissons le voisinage de chaque sommet v_i selon un graphe des k plus proches voisins (k-NNG). Une arête (v_i, v_j) est ajoutée entre deux sommets v_i et v_j si la distance entre $\mathbf{p}(v_i)$ et $\mathbf{p}(v_j)$ est parmi les k plus proches distances de $\mathbf{p}(v_i)$ aux autres points.

2.2 Pondération du graphe

Une fois que le graphe a été créé, il doit être pondéré. Si nous voulons ignorer la similarité des sommets, la fonction de pondération $w : \mathcal{V} \to [0, 1]$ est initialisée avec $w(v_i, v_j) = 1$. Pour prendre en compte les similarités des données associées aux sommets du graphe, il est possible d'utiliser une fonction de similarité reposant sur des distances des arêtes. Soit une fonction initiale $\mathbf{f}^0 : \mathcal{V} \to \mathbb{R}^m$. Calculer les distances entre les sommets correspond à comparer un vecteur caractéristiques $\boldsymbol{\mathcal{P}}(v_i) \in \mathbb{R}^q$ de $\mathbf{f}^0$ tel que :

$$w(v_i, v_j) = exp\left(-\frac{||\boldsymbol{\mathcal{P}}(v_i) - \boldsymbol{\mathcal{P}}(v_j)||_2^2}{\sigma^2}\right). \quad (1)$$

Traditionnellement, on a simplement $\boldsymbol{\mathcal{P}}(v_i) = \mathbf{f}^0(v_i)$. En traitement d'images, un vecteur caractéristique intéressant repose sur la notion de patches [ABM10]. Dans [LEL14], nous avons proposé une nouvelle définition des patches qui peut être utilisée sur n'importe quelle représentation de graphe associé à des maillages ou nuages de points.

2.2.1 Définition du patch

Autour de chaque sommet, nous construisons une grille 2D (le patch). Cette grille est définie sur le plan tangent d'un point (le sommet). Ensuite ce patch est orienté, puis rempli selon une moyenne pondérée de la valeur du signal associé aux sommets voisins.

Orientation : L'algorithme d'estimation de l'orientation du patch, commence par déduire un vecteur tangent $\mathbf{t}_1(v_i)$ à partir de la normale $\mathbf{n}(v_i)$. Nous utilisons une ACP locale sur les coordonnées p_i pour estimer le vecteur normale $\mathbf{t}_0(v_i) = \mathbf{n}(v_i)$. Soit $\mathbf{x}, \mathbf{y}, \mathbf{z}$ les trois axes d'un espace 3D, le premier vecteur tangent $\mathbf{t}_1(v_i)$ est calculé avec :

$$\begin{cases} \mathbf{t}_1(v_i) = & \mathbf{z} \times \mathbf{t}_0(v_i) \text{ si } |\mathbf{z} \cdot \mathbf{t}_0(v_i)| \neq 1 \\ \mathbf{t}_1(v_i) = & \mathbf{x} \times \mathbf{t}_0(v_i) \text{ sinon,} \end{cases} \tag{2}$$

avec $\times$ le produit croisé, et $\cdot$ le produit scalaire. Ensuite le vecteur bitangent $\mathbf{t}_2(v_i)$ est calculé avec $\mathbf{t}_2(v_i) = \mathbf{t}_0(v_i) \times \mathbf{t}_1(v_i)$. Les vecteurs d'orientations $\mathbf{o}_0(v_i), \mathbf{o}_1(v_i), \mathbf{o}_2(v_i)$ sont ensuite respectivement associés à $\mathbf{t}_1(v_i), \mathbf{t}_2(v_i), \mathbf{t}_0(v_i)$.

Construction du patch : La seconde étape consiste à construire les patches. Étant donné un point $\mathbf{p}_i$, un patch est défini en ce point en construisant une grille carré autour de $\mathbf{p}_i$ sur son plan tangent dans l'orientation du patch défini par $(\mathbf{o}_0, \mathbf{o}_1)$. La longueur du patch l est choisie manuellement. Soit n le nombre de cellules sur une ligne du patch. La grille de n^2 cellules est construite autour de $\mathbf{p}_i$ selon la base obtenue à partir du calcul de l'orientation. Ensuite, chaque voisin v_j de v_i est projeté sur le plan tangent de $\mathbf{p}(v_i)$, le point résultant de la projection est noté $\mathbf{p}'(v_j)$. Pour remplir le patch de valeurs, les points projetés $\mathbf{p}'(v_j)$ sont affectés au centre de la cellule la plus proche. La valeur de chaque cellule est ensuite calculée à partir de la moyenne des valeurs $f^0(v_j)$ associées aux sommets v_j des centres des cellules. Cette valeur peut être une valeur spatiale, ou une valeur spectrale (couleurs des points). L'ensemble des valeurs à l'intérieur d'un patch de sommet v_i est noté $\boldsymbol{\mathcal{P}}(v_i)$.

3 Traitement de signaux sur graphes

Soit un graphe pondéré $\mathcal{G} = (\mathcal{V}, \mathcal{E}, w)$ constitué d'un ensemble fini $\mathcal{V} = \{v_1, \ldots, v_N\}$ de N sommets et d'un ensemble fini $\mathcal{E} \subset \mathcal{V} \times \mathcal{V}$ d'arêtes pondérées. Nous supposons $\mathcal{G}$ non dirigé, sans boucle, et sans arêtes multiples. Soit (u, v) une arête de $\mathcal{E}$ connectant les sommets u et v de $\mathcal{V}$. Son poids, noté par $w(u, v)$, représente la similarité entre ces sommets. Ces similarités sont calculées en utilisant une fonction symétrique positive $w : \mathcal{V} \times \mathcal{V} \to \mathbb{R}^+$ satisfaisant $w(u, v) = 0$ si $(u, v) \notin \mathcal{E}$. La notation $u \sim v$ est utilisée pour préciser que les sommets u et v sont adjacents. $\mathcal{H}(\mathcal{V})$ représente un espace de Hilbert des fonctions à valeur réelle définies sur des sommets du graphe. Nous définissons la bordure interne d'un ensemble $A \subset \mathcal{V}$ comme $\partial^- A = \{v_i \in A | \exists v_j \sim v_i, v_j \in \partial A\}$.

De manière similaire, nous définissons $\mathcal{H}(\mathcal{E})$ comme l'espace des fonctions définies sur l'ensemble des arêtes $\mathcal{E}$. Étant donné une fonction $f : \mathcal{V} \to \mathbb{R}$, ses normes ℓ_p et ℓ_∞ sont données par

$$||f||_p = \left(\sum_{v_i \in \mathcal{V}} |f(v_i)|^p \right)^{1/p}, \quad \text{avec } 1 \leq p < \infty, \tag{3}$$

$$||f||_\infty = \max_{v_i \in \mathcal{V}} |f(v_i)|, \quad \text{pour } p = \infty. \tag{4}$$

Soit $\mathcal{G} = (\mathcal{V}, \mathcal{E}, w)$ un graphe pondéré et $w : \mathcal{V} \times \mathcal{V} \to \mathbb{R}^+$ une fonction de poids dépendant de l'interaction entre les sommets. L'*opérateur de difference* [ELB08], dénoté $d_w : \mathcal{H}(\mathcal{V}) \to \mathcal{H}(\mathcal{E})$, est défini pour tout $f \in \mathcal{H}(\mathcal{V})$ et $(v_i, v_j) \in \mathcal{E}$ par $(d_w f)(v_i, v_j) = \sqrt{w(v_i, v_j)}(f(v_j) - f(v_i))$. L'*opérateur de gradient pondéré* d'une fonction $f \in \mathcal{H}(\mathcal{V})$, en un sommet $v_i \in \mathcal{V}$, est le vecteur défini par

$$(\nabla_w f)(v_i) = ((d_w f)(v_i, v_j))^T_{v_j \in \mathcal{V}}. \tag{5}$$

La norme ℓ_p en un sommet v_i est définie, pour $p \geq 1$, comme

$$\|(\nabla_w f)(v_i)\|_p = \left(\sum_{v_j \sim v_i} w(v_i, v_j)^{p/2} \big| f(v_j) - f(v_i) \big|^p \right)^{1/p}. \tag{6}$$

Les *opérateurs de différences morphologiques directionnels* externes et internes sont définis dans [TEL11] par $((d_w f)(v_i, v_j))^\pm$, avec $(x)^+ = \max(x, 0)$ et $(x)^- = (-x)^+$. Leurs gradients pondérés sont :

$$(\nabla_w^\pm f)(v_i) = \Big(((d_w f)(v_i, v_j))^\pm \Big)^T_{v_j \in \mathcal{V}}. \tag{7}$$

Les normes ℓ_p et ℓ_∞ sur ces gradients sont

$$\|(\nabla_w^\pm f)(v_i)\|_p = \left[\sum_{v_j \sim v_i} w(v_i, v_j)^{\frac{p}{2}} \Big((f(v_j) - f(v_i))^\pm \Big)^p \right]^{\frac{1}{p}}, \tag{8}$$

$$\|(\nabla_w^\pm f)(v_i)\|_\infty = \max_{v_j \sim v_i} \sqrt{w(v_i, v_j)}(f(v_j) - f(v_i))^\pm). \tag{9}$$

L'*opérateur p-Laplacien pondéré isotrope*, avec $p \in [1, +\infty[$, en un sommet $v_i \in \mathcal{V}$ est défini sur $\mathcal{H}(\mathcal{V})$ par [ELB08] comme

$$(\Delta_{w,p}^i f)(v_i) = \sum_{v_j \sim v_i} \psi_{w,p}^i (v_i, v_j)(f(v_i) - f(v_j)), \tag{10}$$

où

$$\psi_{w,p}^i(v_i, v_j) = \frac{1}{2} w(v_i, v_j)(||\nabla_w f(v_i)||_2^{p-2} + ||\nabla_w f(v_j)||_2^{p-2}). \tag{11}$$

L'∞-*Laplacien* est défini par [EDL12] comme

$$(\Delta_{w,\infty} f)(v_i) = \frac{1}{2}\left[||(\nabla_w^+ f)(v_i)||_\infty - ||(\nabla_w^- f)(v_i)||_\infty||\right]. \tag{12}$$

Soit une fonction générale $f^0 : \mathcal{V} \subset \mathbb{R}^n \to \mathbb{R}^m$ définie sur un graphe de topologie arbitraire, correspondant à l'observation d'une fonction $h \in \mathcal{H}(\mathcal{V})$ bruitée par un bruit additif $n \in \mathcal{H}(\mathcal{V})$ tel que $f^0 = h + n$. Régulariser f^0 correspond à un problème d'optimisation qui peut être formalisé par la minimisation d'une énergie représentée par la somme pondérée d'un terme de régularisation avec un terme d'attache aux données :

$$h \approx \underset{f:\mathcal{V}\to\mathbb{R}}{\arg\min}\ J^\phi_{w,p}(f) + \frac{\lambda}{2}||f - f^0||_2^2 \tag{13}$$

$$\text{où} \quad J^\phi_{w,p}(f) = \textstyle\sum_{v_i\in\mathcal{V}} \phi(||(\nabla_w f)(v_i)||_p). \tag{14}$$

Pour résoudre (14), nous considérons l'équation de Euler-Lagrange :

$$\frac{\partial J^\phi_{w,p}(f)}{\partial f(v_i)} + \lambda(f(v_i) - f^0(v_i)) = 0, \forall v_i \in \mathcal{V}. \tag{15}$$

Dans [LEL14], nous avons montré que la solution de (15) peut être obtenue avec l'algorithme itératif suivant :

$$f^{n+1}(v_i) = \frac{\lambda f^0(v_i) + \sum_{v_j\sim v_i} \beta^{\phi,p,f}_{v_i,v_j} f^n(v_j)}{\lambda + \sum_{v_j\sim v_i} \beta^{\phi,p,f}_{v_i,v_j}}. \tag{16}$$

avec

$$\beta^{\phi,p,f}_{v_i,v_j} = \alpha^{\phi,p,f}_{v_i,v_j} |f(v_j) - f(v_i)|^{p-2} \tag{17}$$

et

$$\alpha^{\phi,p,f}_{v_i v_j} = w(v_i, v_j)^{p/2} \left(\frac{\phi'(||(\nabla_w f)(v_i)||_p)}{||(\nabla_w f)(v_i)||_p^{p-1}} + \frac{\phi'(||(\nabla_w f)(v_j)||_p)}{||(\nabla_w f)(v_j)||_p^{p-1}} \right). \tag{18}$$

L'équation 16 décrit une famille de traitements de diffusion sur graphe, paramétrée par la structure du graphe (sa topologie et la fonction de poids w) et le paramètre λ (paramètre de fidélité). Dans la suite nous allons considérer notre nuage de points comme un graphe pondéré où la fonction $w(u, v)$ est une fonction d'interaction entre deux patchs que nous allons définir par la suite.

4 Applications

Nous illustrons la méthode proposée pour traiter des données sur nuage de points (coordonnées ou couleurs). Étant donné un graphe pondéré $\mathcal{G} = (\mathcal{V}, \mathcal{E}, w)$ associé à un nuage de points, nous considérons une fonction sur graphe initiale $\mathbf{f}^0 : \mathcal{V} \to \mathbb{R}^m$ avec $m = 3$.

4.1 Colorisation

La colorisation est le processus d'ajouter des couleurs sur un objet non coloré. Soit $\mathbf{f}^0 : \mathcal{V} \to \mathcal{C}$ une fonction qui assigne des couleurs à des sommets. Soit $A \subset \mathcal{V}$ le sous-ensemble de sommets avec des couleurs non connues et ∂A le sous-ensemble de sommets avec les valeurs connues. Le problème de la colorisation revient à un problème d'interpolation pour trouver la fonction $\hat{f}$ approximant f^0 de $\mathcal{V}$ qui minimise l'énergie suivante :

(a) Bas-relief incolore avec des annotations (b) Bas-relief coloré (c) Original (d) Resultat

FIGURE 1: (a) et (b), colorisation d'un bas-relief (de 506k points). (c) et (d) simplification d'un nuage de point de la statue de Saint Eligius après 10k itérations.

$$\hat{f} = \arg \min_{f\in H(\mathcal{V})} \left\{ R_{w,p}(f) = \sum_{v_i\in A} ||\nabla_w f(v_i)||_p^p \right\}, \tag{19}$$

avec $f(v_i) = f^0(v_i)$, pour $v_i \in \partial A$. Le problème peut être résolu en utilisant les équations d'Euler-Lagrange :

$$\begin{cases} (\Delta^i_{w,p} f)(v_i) = 0 & \forall v_i \in A, \\ f(v_i) = f^0(v_i) & \forall v_i \in \partial A. \end{cases} \tag{20}$$

La solution de (20) est obtenue avec l'algorithme itératif de (16) avec $\lambda = 0, \forall v_i \in \partial^- A$. La fonction de similarité $w : \mathcal{V}\times\mathcal{V} \to \mathbb{R}$ est calculé à partir de la comparaison de patches des vecteurs caractéristiques géométriques. La colorisation commence à partir des points qui sont voisins des points annotés et croit à travers les itérations. L'algorithme itératif s'arrête lorsque l'ensemble des points colorés est égal à l'ensemble vide. La Figure 1 montre la colorisation d'un bas-relief.

4.2 Filtrage et simplification

Soit $\mathbf{f}^0 : \mathcal{V} \to \mathcal{P}$ les coordonnées des sommets. L'algorithme itératif de (16) permet de filtrer la géométrie des objets 3D en utilisant $\lambda = 0$. La Figure 1 montre la simplification d'un nuage de points avec une réduction du nombre de points de 77.6% (de 201k points à 45k points) en utilisant les paramètres $p = 0.1$ et $\phi(s) = s^p$.

4.3 Inpainting

L'inpainting consiste à déduire des valeurs manquantes à partir des valeurs connues. Nous considérons des données définies sur un domaine général représenté par un graphe $\mathcal{G} = (\mathcal{V}, \mathcal{E}, w)$. Soit $\mathbf{f}^0 : \mathcal{V} \to \mathcal{C}$ une fonction qui assignent une couleur aux sommets. Soit $\mathcal{A} \subset \mathcal{V}$ le sous-ensemble des sommets de valeurs inconnues et $\partial\mathcal{A}$ le sous-ensemble de sommets de valeurs

(a) Vase bruité (b) Vase restauré

FIGURE 2: Restauration des informations géométriques et couleur d'un vase.

connues. Le but de l'interpolation est de trouver une fonction $\hat{f}$ approximant $f^0 \in \mathcal{V}$ et correspond à résoudre :

$$\begin{cases} (\Delta_{w,\infty} f)(v_i) = 0 & \forall v_i \in \mathcal{A}, \\ f(v_i) = f^0(v_i) & \forall v_i \in \partial\mathcal{A}. \end{cases} \tag{21}$$

Les travaux de [LEL14] ont montré que le problème d'interpolation a une solution unique qui peut être obtenu avec l'algorithme itératif présenté dans [EDL12]. À chaque fin d'itération, l'ensemble $\partial\mathcal{A}$ est mise à jour avec $\partial\mathcal{A}^{(n+1)} = \partial\mathcal{A}^{(n)} \cup \partial^-\mathcal{A}^{(n)}$ et $\partial^-\mathcal{A}^{(n+1)}$ est mise à jour avec $\partial\mathcal{A}^{(n+1)}$. L'algorithme s'arrête lorsque l'ensemble des sommets à interpoler est vide. La Figure 2 montre la restauration d'un vase cassé. La partie géométrique est d'abord corrigée, ensuite la couleur est interpolée.

4.4 Segmentation

La segmentation consiste à partitionner un objet 3D en plusieurs régions. Soit $f : \mathcal{V} \to \mathbb{R}$ une fonction sur un graphe $\mathcal{G}(\mathcal{V}, \mathcal{E}, w)$ à segmenter. Le problème de la segmentation peut être formulé par une EDP à résoudre sur des graphes pondérés. Cette dernière formulation repose sur la propagation de front de l'équation eikonale pour calculer des distances généralisées sur graphe. Soit $L = \{l_1, ..., l_n\}$ l'ensemble des labels. L'objectif de la propagation de label, est de labéliser chaque sommet $u \in \mathcal{V}$ sous la condition que u soit un voisin de même label. La propagation de label peut être formalisé par la résolution de l'équation eikonale sur un graphe G tel que :

$$\begin{cases} ||(\nabla_w^- \phi)(v_i, t)||_2 = 1 & v_i \in \mathcal{V} \\ f(v_i) = \phi(v_i) & x \in \gamma \subset \mathcal{V} \end{cases} \tag{22}$$

où $\phi : \mathcal{V} \to \mathbb{R}$ représente l'ensemble des labels initiaux. La résolution de l'équation eikonale (22) sur un graphe pondéré est étudiée dans [DEL13]. La fonction de similarité $w : \mathcal{V} \times \mathcal{V} \to \mathbb{R}^+$ prend en compte la distance colorimétrique $w(v_i, v_j) = e^{-||\mathbf{f}^0(v_i) - \mathbf{f}^0(v_j)||/\sigma^2}$, où $\mathbf{f}^0 : \mathcal{V} \to \mathcal{C}$ sont les couleurs associées au noeud $v_i \in \mathcal{V}$. La Figure 3 montre la segmentation de quelques nuages de points après la résolution de l'équation eikonale.

(a) Vase labelélisé (b) Résultat (c) Vase labelelisé (d) Résultat

FIGURE 3: Segmentation d'un vase en résolvant l'équation eikonale sur graphe.

5 Conclusion

Ce papier propose une approche pour traiter des fonctions sur nuage de points 3D représentées par des graphes. Nous avons montré comment traduire des EDPs en utilisant le cadre des EdPs. Nous avons appliqué cette approche sur des nuages de points en lien avec le patrimoine culturel pour résoudre des problèmes inverses tels que la restauration ou l'interpolation.

Références

[ABM10] A.Buades, B.Coll, and J.-M. Morel. Image denoising methods. a new nonlocal principle. SIAM Rev., 52(1) :113–147, 2010.

[AJW13] Patrice Abry, Stéphane Jaffard, and Herwig Wendt. When van gogh meets mandelbrot : Multifractal classification of painting's texture. Signal Process., 93(3) :554–572, 2013.

[DEL13] X. Desquesnes, A. Elmoataz, and O. Lézoray. Eikonal equation adaptation on weighted graphs : fast geometric diffusion process for local and non-local image and data processing. J. Math. Imaging Vis., 46(2) :238–257, 2013.

[EDL12] A. Elmoataz, X. Desquesnes, and O. Lezoray. Nonlocal morphological pdes and p -laplacian equation on graphs with applications in image processing and machine learning. IEEE J. Sel. Top. Signa., 6(7) :764 –779, 2012.

[ELB08] A. Elmoataz, O. Lezoray, and S. Bougleux. Nonlocal discrete regularization on weighted graphs : a framework for image and manifold process ing. IEEE T. Image Process., 17(7) :1047–1060, 2008.

[LEL14] F. Lozes, A. Elmoataz, and O. Lezoray. Partial difference operators on weighted graphs for image processing on surfaces and point clouds. IEEE T. Image Process., 23(9) :3896–3909, Sept 2014.

[TEL11] V.-T. Ta, A. Elmoataz, and O. Lézoray. Nonlocal pdes-based morphology on weighted graphs for image and data processing. IEEE T. Image Process., 26(2) :1504–1516, june 2011.