Méthodes denses d'interpolation de mouvement pour le codage vidéo distribué monovue et multivue

Thomas MAUGEY, Wided MILED, Marco CAGNAZZO, Béatrice PESQUET-POPESCU
TELECOM ParisTech, CNRS LTCI, Département de Traitement du Signal et des Images
46 rue Barrault, 75634 Paris Cedex 13, France
{maugey, miled, cagnazzo, pesquet}@telecom-paristech.fr

Résumé – Les travaux présentés dans cet article traitent d'un ensemble de méthodes d'interpolation d'images appliquées au contexte du codage vidéo distribué monovue et multivue. Ces méthodes, fondées sur des champs de vecteurs denses, peuvent être utilisées aussi bien pour des estimations dans le sens des vues que pour des estimations dans le sens temporel. Nous comparons, dans cet article, ces méthodes denses à celles par bloc classiquement utilisées en codage vidéo distribué.

Abstract – This work is about a set of methods for dense image interpolation in the framework of monoview and multiview distributed video coding. The interpolation techniques can be used for multiview video, both in the case of temporal and in the case of inter-view interpolation. The proposed dense disparity and motion estimation methods have been compared to classical block-based ones by carrying out several experiments whose results validate the dense vector approach.

1 Introduction

Plusieurs solutions pratiques du codage vidéo distribué, permettant de supprimer l'étude de la corrélation entre trames à l'encodeur, sont apparues il y a quelques années. Une de ces approches, proposée par Stanford [1], est celle adoptée dans ces travaux. Dans le schéma de codage correspondant, la séquence vidéo est divisée en des trames clé (TC) et des trames Wyner-Ziv (TWZ). Les trames clés sont transmises grâce à un codage intra (par exemple H.264 intra) et sont utilisées au décodeur pour générer une estimation des trames WZ, appelée information adjacente. Cette estimation est corrigée par les bits de parités générés par un turbo-encodage des TWZ. Les performances de ce schéma de codage dépendent fortement de la qualité de l'information adjacente : plus elle est proche de la TWZ originale, moins le codeur a besoin de transmettre de bits de parités pour la corriger.

Dans un schéma de codage multivue, l'extraction de l'information adjacente au décodeur utilise le champ de mouvement entre la trame d'avant et la trame d'après pour l'interpolation temporelle et le champ de disparité entre la trame gauche et la trame droite pour l'interpolation intervues. Comme ces champs de vecteurs sont estimés au décodeur et qu'il n'y a pas besoin de les transmettre comme dans un schéma de codage classique, nous proposons d'utiliser une technique d'estimation dense de ces vecteurs de déplacement. Deux méthodes d'estimation dense sont ainsi utilisées dans l'objectif d'améliorer la qualité de l'information adjacente par rapport à la méthode de référence DISCOVER, qui s'avère être l'une des plus performantes actuellement. L'idée de cette amélioration est de maintenir la structure d'interpolation de DISCOVER et de rajouter deux blocs de raffinement en utilisant l'une des méthodes d'estimation de champs de déplacement denses. La première méthode est fondée sur l'algorithme de Cafforio-Rocca [2] et la deuxième est fondée sur l'approche variationnelle convexe décrite dans [3].

Cet article est structuré comme suit. La section 2 décrit la structure générale de la méthode d'interpolation proposée. Nous détaillerons ensuite dans les section 3 et 4 les deux méthodes de raffinement utilisées. Les résultats expérimentaux sont présentés dans la section 5, et la conclusion est enfin donnée à la section 6.

2 Structure de l'algorithme proposé

L'algorithme d'interpolation proposé dans le cadre de ce travail est fondé sur la structure du codeur DISCOVER, auquel on rajoute deux blocs de raffinement effectuant une estimation dense et précise des champs de vecteur. Le schéma général de l'algorithme est représenté dans la figure 1. Les blocs en traits pleins correspondent à ceux existants dans la méthode DISCOVER, et ceux en traits pointillés sont ceux que nous nous proposons de rajouter. Les deux images en entrée, notées I_1 et I_2, correspondent à deux trames clés acquises à deux instants différents ou de deux points de vues différents. Après un filtrage spatial passe-bas, un premier champ de vecteur $\mathbf{u}^a$ entre I_1 et I_2 est généré grâce à une estimation monodirectionnelle (MD), basée sur une simple opération de recherche par bloc. Ensuite une opération de raffinement est effectuée dont il résulte un champ $\mathbf{u}^b = f_1(\mathbf{u}^a)$. Le champ $\mathbf{u}^b$ est utilisé ensuite, par l'estimateur bidirectionnel (BD) de DISCOVER, pour générer les champs de vecteurs, $\mathbf{u}_1^c$ et $\mathbf{u}_2^c$, entre la TWZ et les deux TC. Ces champs sont régularisés à l'aide d'un filtre médian. Un dernier

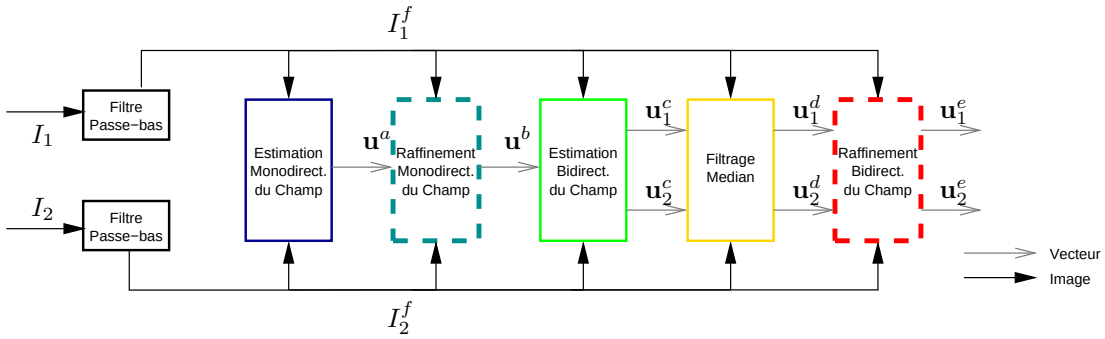

FIG. 1 – Schéma général des méthodes d'interpolation envisagées.

raffinement, f_2, est alors effectué sur ces champs produisant le couple de champs final, $\mathbf{u}_1^e$ et $\mathbf{u}_2^e$.
Pour chacune des deux étapes de raffinement f_1 et f_2, trois traitements sont envisageables. Le premier est de ne rien faire, ce qui nous ramène à la méthode initiale de DISCOVER. Le second est d'utiliser l'algorithme de raffinement de Cafforio-Rocca (CR) expliqué dans la section 3, et le troisième est d'utiliser la méthode variationnelle (VT) expliquée dans la section 4. En conclusion, neuf méthodes sont envisageables, dont une est celle de DISCOVER que nous noterons (D). Dans ces travaux, nous commençons l'exploration de ces méthodes en considérant uniquement celles caractérisées par une seule étape de raffinement. On appellera CR-1 et CR-2 [VT-1 et VT-2] les techniques effectuant les raffinements f_1 et f_2, respectivement, par la méthode CR [VT].

3 L'algorithme de Cafforio-Rocca modifié

L'algorithme de Cafforio-Rocca (ACR) [2] permet d'estimer un champ de vecteur de mouvement ou un champ de disparité entre deux images appelées I_1 et I_2. Pour chaque position $\mathbf{p}$, l'ACR s'effectue en trois étapes :

Initialisation. Le vecteur courant, $\mathbf{u}^{(1)}(\mathbf{p})$, est initialisé (souvent avec le dernier vecteur estimé, un vecteur voisin).

Validation. On confronte l'erreur de prédiction associée à $\mathbf{u}^{(1)}$ avec l'erreur non compensée. Si l'erreur compensée dépasse de plus de γ l'erreur non compensée, le vecteur courant est mis à zero : $\mathbf{u}^{(2)} = \mathbf{0}$. Dans le cas contraire on garde : $\mathbf{u}^{(2)} = \mathbf{u}^{(1)}$.

Correction. Le vecteur validé est modifié en y ajoutant une correction $\delta\mathbf{u}$, qui est obtenue en minimisant l'erreur de prédiction, sous une contrainte sur la longueur. La correction est choisie de manière à minimiser :

$$J(\delta\mathbf{u}) = [I_k(\mathbf{p}) - I_{k-1}(\mathbf{p}+\mathbf{u}^{(2)}+\delta\mathbf{u})]^2 + \lambda\|\delta\mathbf{u}\|^2. \quad (1)$$

En utilisant un développement au premier ordre, on peut trouver la correction optimale :

$$\delta\mathbf{u}(\mathbf{p}) = \frac{-\epsilon\boldsymbol{\varphi}}{\lambda + \|\boldsymbol{\varphi}\|^2},$$

où $\epsilon = I_k(\mathbf{p}) - I_{k-1}(\mathbf{p}+\mathbf{u}^{(2)})$ est l'erreur de prédiction associée à $\mathbf{u}^{(2)}$, et $\boldsymbol{\varphi} = \nabla I_2(\mathbf{p}+\mathbf{u}^{(2)})$ est le gradient compensé de l'image de réference. Le vecteur final est donc $\mathbf{u}^{(3)} = \mathbf{u}^{(2)} + \delta\mathbf{u}$.

Cet algorithme peut être utilisé dans le schéma d'interpolation de la Fig. 1 pour effectuer le raffinement monodirectionnel (MD) ou bidirectionnel (BD). Dans les deux cas, on a en entrée les deux images de référence I_1 et I_2. En outre, la version MD a en entrée un seul champ de vecteur, $\mathbf{u}^a$, qui a été calculé par l'estimateur MD de DISCOVER, tandis que la version BD a en entrée les deux champs, $\mathbf{u}_1^d$, $\mathbf{u}_2^d$, calculés par l'estimateur BD. Pour les deux versions du raffinement, les vecteurs en entrée sont traités par les trois étapes de l'ACR modifié comme suit :

1. On utilise le vecteur [les vecteurs] d'entrée pour initialiser le premier vecteur [les premiers vecteurs] d'un bloc. Ensuite, les autres vecteurs d'un même bloc sont initialisés par une combinaison linéaire des vecteurs voisins.
2. Pour la validation, on calcule l'erreur compensée, l'erreur non compensée, ainsi que l'erreur associée au vecteur d'entrée :
$$A = \left|I_1(\mathbf{p}) - I_2(\mathbf{p}+\mathbf{u}^{(1)}(\mathbf{p}))\right|$$
$$B = |I_1(\mathbf{p}) - I_2(\mathbf{p})| + \gamma$$
$$C = |I_1(\mathbf{p}) - I_2(\mathbf{p}+\mathbf{u}^a)|$$
ou, dans le cas bidirectionnel :
$$A = \left|I_1(\mathbf{p}+\mathbf{u}_1^{(1)}(\mathbf{p})) - I_2(\mathbf{p}+\mathbf{u}_2^{(1)}(\mathbf{p}))\right|$$
$$B = |I_1(\mathbf{p}) - I_2(\mathbf{p})| + \gamma$$
$$C = \left|I_1(\mathbf{p}+\mathbf{u}_1^d) - I_2(\mathbf{p}+\mathbf{u}_2^d)\right|.$$
On choisi le vecteur associé qui présente l'erreur la plus faible.
3. Pour le raffinement, on introduit de nouvelles fonctions de coût. Pour le cas MD, on utilise la matrice de diffusion $\mathbf{D}$ qui assure une régularité spatiale des vecteurs :
$$\mathbf{D}(\nabla I_2) = \frac{1}{|\nabla I_2|^2 + 2\sigma^2}\left[\begin{pmatrix}\frac{\partial I_2}{\partial y}\\ -\frac{\partial I_2}{\partial x}\end{pmatrix}\begin{pmatrix}\frac{\partial I_2}{\partial y}\\ -\frac{\partial I_2}{\partial x}\end{pmatrix}^T + \sigma^2\mathbf{I}_2\right].$$
$\mathbf{I}_2$ étant la matrice identité de taille 2×2 et σ une constante qui contrôle le poids de la diffusion anisotrope. La fonction de coût devient :
$$J(\delta\mathbf{u}) = [I_1(\mathbf{p}) - I_2(\mathbf{p}+\mathbf{u}^{(2)}+\delta\mathbf{u})]^2 + \lambda\delta\mathbf{u}^T\mathbf{D}\delta\mathbf{u}.$$

L'opérateur $\mathbf{D}$, dit de Nagel et Enkelmann [4], contrôle le lissage des vecteurs tout en autorisant ses discontinuités au niveau des variations importantes d'intensités, c'est à dire lorsque $|\nabla I_2| \gg \sigma$. La diffusion de cet opérateur dépend donc du choix de paramètre σ. Si sa valeur est élevée, alors la diffusion est isotrope partout et se ramène à celle de Tikhonov, et donc de l'algorithme original, et si σ est choisi faible, l'opérateur $\mathbf{D}$ opère de manière uniformément anisotrope.

Nous avons résolu le problème de minimisation de la nouvelle fonction de coût. Le raffinement optimal associé s'écrit :

$$\delta\mathbf{u}^* = \frac{-\epsilon\mathbf{D}^{-1}\boldsymbol{\varphi}}{\lambda + \boldsymbol{\varphi}^T\mathbf{D}^{-1}\boldsymbol{\varphi}}.$$

Dans le cas BD, on prend en compte les deux corrections simultanément. On obtient ainsi la fonction de coût suivante :

$$\begin{aligned} J(\delta\mathbf{u}_1, \delta\mathbf{u}_2) = [&I_1(\mathbf{p} + \mathbf{u}_1^{(1)} + \delta\mathbf{u}_1) \\ &- I_2(\mathbf{p} + \mathbf{u}_2^{(1)} + \delta\mathbf{u}_2)]^2 \\ &+ \lambda_1\|\delta\mathbf{u}_1\|^2 + \lambda_2\|\delta\mathbf{u}_2\|^2, \end{aligned} \quad (2)$$

et les raffinements optimaux associés :

$$\delta\mathbf{u}_1^* = \frac{-\epsilon\boldsymbol{\varphi}_1}{\lambda_1 + \|\boldsymbol{\varphi}_1\|^2 + \|\boldsymbol{\varphi}_2\|^2} \quad (3)$$

$$\delta\mathbf{u}_2^* = \frac{\epsilon\boldsymbol{\varphi}_2}{\lambda_2 + \|\boldsymbol{\varphi}_1\|^2 + \|\boldsymbol{\varphi}_2\|^2}, \quad (4)$$

où ϵ désigne ici est l'erreur bidirectionelle, et $\boldsymbol{\varphi}_1$, $\boldsymbol{\varphi}_2$ les gradients compensés associés aux deux images :

$$\begin{aligned} \boldsymbol{\varphi}_1 &= \nabla I_1(\mathbf{p} + \mathbf{u}_1^{(1)}) \\ \boldsymbol{\varphi}_2 &= \nabla I_1(\mathbf{p} + \mathbf{u}_2^{(1)}). \end{aligned}$$

4 Approche variationnelle convexe

La deuxième approche que nous avons considérée dans ce travail a été proposée dans [3] pour l'estimation dense de la disparité entre deux images rectifiées. Cependant, elle peut être facilement étendue pour l'estimation 2D de champs de mouvement ou de disparité, avec une géométrie quelconque du capteur stéréoscopique. Cette approche variationnelle formule le problème d'estimation dense comme la minimisation d'une fonctionnelle J_v qui s'exprime généralement comme suit :

$$J_v(\mathbf{u}) = \sum_{\mathbf{p}\in\mathcal{D}} [I_2(\mathbf{p}) - I_1(\mathbf{p} + \mathbf{u})]^2 \quad (5)$$

où $\mathcal{D} \subset \mathbb{N}^2$ est le support de l'image. Cependant, la minimisation de cette fonction, non-convexe par rapport à $\mathbf{u}$, ne permet de converger que vers des minima locaux. Pour contourner cette difficulté, nous supposons qu'une estimée initiale $\bar{\mathbf{u}}$ de $\mathbf{u}$ est accessible et nous développons, autour de cette estimée, le terme non-linéaire en série de Taylor au premier ordre comme suit :

$$I_1(\mathbf{p} + \mathbf{u}) \simeq I_1(\mathbf{p} + \bar{\mathbf{u}}) + (\mathbf{u} - \bar{\mathbf{u}})\ \nabla I_1(\mathbf{p} + \bar{\mathbf{u}}) , \quad (6)$$

où $\nabla I_1(\mathbf{p} + \bar{\mathbf{u}})$ est le gradient de l'image I_1 compensée par le champ de vecteurs $\bar{\mathbf{u}}$. En introduisant la linéarisation ci-dessus, l'expression du critère (5) se réécrit de la manière suivante :

$$J_v(\mathbf{u}) = \sum_{\mathbf{p}\in\mathcal{D}} [\mathbf{L}(\mathbf{p}).\ \mathbf{u}(\mathbf{p}) - \mathbf{r}(\mathbf{p})]^2 , \quad (7)$$

avec
$$\begin{aligned} &\mathbf{L}(\mathbf{p}) = \nabla I_1(\mathbf{p} + \bar{\mathbf{u}}(\mathbf{p})), \\ &\mathbf{r}(\mathbf{p}) = I_2(\mathbf{p}) - I_1(\mathbf{p} + \bar{\mathbf{u}}(\mathbf{p})) + \bar{\mathbf{u}}(\mathbf{p})\ \mathbf{L}(\mathbf{p}). \end{aligned}$$

Minimiser le critère J_v est un problème inverse consistant à estimer le champ de déplacement inconnu $\mathbf{u}$ à partir des champs d'observation $\mathbf{L}$ et $\mathbf{r}$. La difficulté pour résoudre ce problème *mal posé* réside dans le fait que les composantes de $\mathbf{L}$ peuvent s'annuler en certains points de l'image. Pour obtenir des solutions uniques et fiables, il faut donc incorporer autant d'information *a priori* que possible sur le champ de vecteurs à estimer.

Le problème d'optimisation sous contraintes ainsi obtenu a été formulé dans un cadre ensembliste [3] où chaque contrainte introduite sur la solution est modélisée comme un ensemble de niveau d'une fonction convexe. L'intersection de tous ces ensembles convexes et fermés constitue l'ensemble des solutions admissibles [5]. La première contrainte que nous avons imposée est la contrainte de plage de valeurs qui limite l'intervalle de déplacement de disparité ou de mouvement. L'ensemble associé à cette information est :

$$S_1 \quad = \{\mathbf{u} \in \mathcal{H} \mid u_{\min} \leq \mathbf{u} \leq u_{\max}\} . \quad (8)$$

Le second *a priori* que nous avons considéré concerne la régularité des champs de mouvement et de disparité, lisses dans les régions homogènes et discontinus aux frontières des objets. Pour traduire cette propriété, nous avons utilisé une contrainte de régularisation basée sur la variation totale (VT), qui est une mesure de la somme des amplitudes des discontinuités présentes dans l'image. La régularisation par variation totale a été appliquée pour résoudre divers problèmes inverses mal posés et s'est particulièrement imposée comme une approche performante en traitement d'images [3], [6]. L'idée de cette régularisation est motivée par l'observation que, dans de nombreux types de problèmes, la variation totale de l'image originale ne dépasse pas une certaine borne connue. Cette information restreint la solution à appartenir à l'ensemble convexe suivant :

$$S_2 \quad = \{\mathbf{u} \in \mathcal{H} \mid \mathrm{VT}(\mathbf{u}) \leq \tau_u\} , \quad (9)$$

où τ_u est une constante positive qui peut être estimée à partir d'expérimentation ou en exploitant des bases de données d'images du même type.

Le problème d'estimation dense du champ de déplacement $\mathbf{u}$ est finalement formulé comme celui de la minimisation de la fonctionnelle (7) sous les contraintes (8) et (9). Pour résoudre numériquement ce problème, nous avons adapté l'algorithme parallèle et itératif par blocs proposé par Combettes [5] pour résoudre des problèmes de restauration d'images. Cet algorithme offre une méthode de résolution puissante et efficace pour l'optimisation d'une fonction convexe et différentiable.

	foreman			mobile		
QP	CR-1	CR-2	VT-1	CR-1	CR-2	VT-1
31	0.49	**0.65**	0.36	1.09	-0.07	**1.47**
34	0.50	0.48	**0.53**	0.84	-0.01	**1.16**
37	**0.48**	0.40	0.31	0.78	0.03	**0.91**
40	**0.39**	0.27	0.25	0.36	0.07	**0.59**

TAB. 1 – Δ_{PSNR} entre l'information adjacente générée par CR-1, CR-2 et VT-1 et celle générée par DISCOVER.

5 Résultats expérimentaux

Les méthodes présentées ont été validées sur des séquences monovue et multivues dont les TC ont été codées en H.264 Intra à quatre niveaux de quantification différents correspondant à des pas (QP) égaux à 31, 34, 37 et 40. La mesure de la qualité des estimations est le PSNR entre les images originales et interpolées. Les résultats sont présentés comme la différence entre le PSNR des méthodes proposées et celui de DISCOVER. Dans la plupart de nos tests, la méthode VT-2 n'améliorait pas la qualité obtenue par DISCOVER. Le raffinement effectué dégradait même parfois l'estimation et donc nous avons choisis de ne pas présenter ses résultats dans la suite.

Les résultats des tests effectués sur les séquences CIF monovue, "foreman" et "mobile" sont présentés dans le tableau 1. On peut voir qu'effectuer le raffinement sur les vecteurs monodirectionnels (CR-1 et VT-1) permet d'obtenir de meilleurs résultats dans la plupart des cas observés, avec des gains jusqu'à 1.47 dB par rapport à DISCOVER.

Dans le cas multivue la séquence testée est "Ballet", de taille 512×384 pixels. Les estimations temporelles générées confirment les résultats obtenus précédemment comme indiqués dans la figure 2, même si les gains sont plus faibles. En ce qui concerne les estimations générées dans le sens des vues, on observe que la méthode VT-1 reste la plus performante. Les trois méthodes présentent cependant des valeurs très proches, qui sont faibles par rapport aux gains obtenus dans le sens temporel. Cela confirme qu'il est plus difficile d'extraire la corrélation dans le sens des vues que dans le sens du temps.

6 Conclusion

Nous avons présenté, dans cette communication, une famille de méthodes d'interpolation d'images fondées sur des champs de vecteurs denses. Nous avons testés quatre d'entre elles dont trois ont montré des résultats supérieurs à l'état de l'art. Par la suite, nous allons continuer les tests pour les quatre méthodes restantes, effectuant un double raffinement.

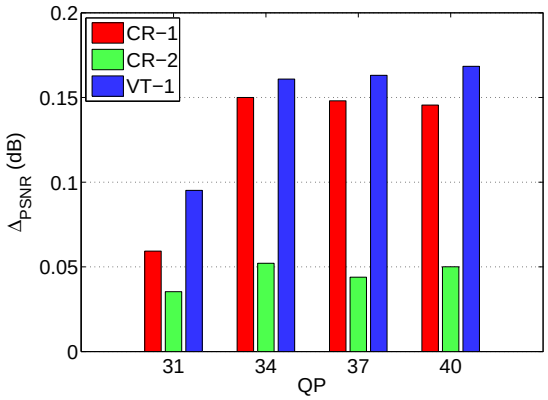

FIG. 2 – Résultats sur la séquence "Ballet", 512×384, dans le sens du temps.

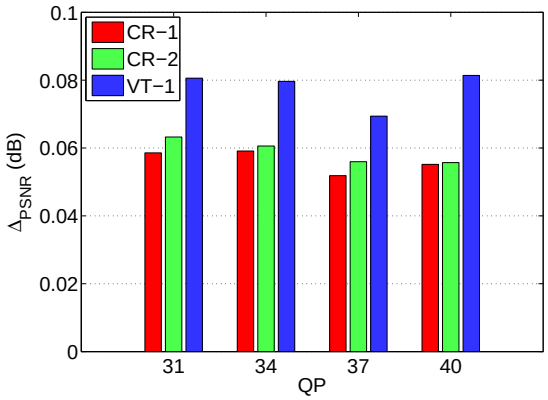

FIG. 3 – Résultats sur la séquence "Ballet", 512×384, dans le sens des vues.

Références

[1] A. Aaron, R. Zhang, and B. Girod, "Wyner-Ziv coding of motion video," in *Asilomar Conference on Signals and Systems*, Pacific Grove, California, Nov. 2002.

[2] Ciro Cafforio and Fabio Rocca, "Methods for measuring small displacements of television images," *IEEE Trans. Inform. Theory*, vol. IT-22, no. 5, pp. 573–579, Sept. 1976.

[3] W. Miled, J.-C. Pesquet, and M. Parent, "Disparity map estimation using a total variation bound," in *Proc. 3rd Canadian Conf. Comput. Robot Vis.*, Quebec, Canada, Jun. 2006, pp. 48–55.

[4] H. Nagel and W. Enkelmann, "An investigation of smoothness constraints for the estimation of displacement vector fields from image sequences," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. PAMI-8, no. 5, pp. 565–593, Sept. 1986.

[5] P. L. Combettes, "A block iterative surrogate constraint splitting method for quadratic signal recovery," *Trans. Signal Process.*, vol. 51, pp. 1771–1782, July 2003.

[6] L. I. Rudin, S. Osher, and E. Fatemi, "Nonlinear total variation based noise removal algorithms," *Physica D*, vol. 51, pp. 259–268, 60.