Détection de flou en temps réel basée sur les motifs binaires locaux

Yannick Faula

Stéphane Bres

Véronique Églin

Université de Lyon, INSA Lyon, LIRIS, UMR5205, F-69621, France

{yannick.faula, stephane.bres, veronique.eglin}@liris.cnrs.fr

1 Résumé

Notre objectif est la détection de flou en temps réel sans référence et sans apprentissage dans des images peu texturées. Cette phase est utile, par exemple, pour la détection de fissures fines qui n'est possible qu'avec une acquisition d'images de qualité suffisante. En effet, une image de mauvaise qualité est difficile à traiter et les détériorations n'ont pas les mêmes conséquences sur les traitements. De manière générale, la phase d'acquisition est la première étape cruciale dans beaucoup de domaines comme la photogrammétrie ou le contrôle de qualité et sa réussite dépend de la qualité des données récoltées. Les acquisitions massives automatiques d'images ne sont pas envisageables si un opérateur humain doit vérifier la qualité de chaque photo. Dans le cas particulier qui nous intéresse ici, celui de la surveillance d'ouvrage d'art, l'effet indésirable le plus contraignant est le flou. Il peut être causé par un mouvement au moment de la prise de photo ou par un effet de mauvaise focalisation (out-of-focus) qui peut survenir très facilement en utilisant une focale fixe qui impose une profondeur de champs réduite. Nous avons élaboré une nouvelle mesure pour la détection de flou sans référence et sans apprentissage, destinée à évaluer en temps réel la qualité d'une image dans un contexte d'images très peu texturées et avec très peu de contraste. Les images de tests sont des images de surfaces de béton issues de la surveillance d'ouvrage d'art. La détection de flou est utilisée pendant l'acquisition et avant la détection de défauts comme les fissures. Nous présentons ici plusieurs méthodes génériques aux problèmes de contrôle qualité des images. Puis nous présenterons notre méthode, dérivée directement du détecteur de fissures FLASH présenté dans [1], afin de réaliser les calculs directement en une seule passe au moment de l'acquisition.

1.1 Etat de l'art

Les méthodes de détection de flou diffèrent par l'utilisation de l'information disponible sur la "meilleure" version d'une image. Par exemple, la "meilleure" version d'une image peut être une image prise à haute résolution et non traitée par un algorithme de compression. Plusieurs techniques sont basées sur l'apprentissage avec les SVM (Support Vector Machines) ou les CNN (Convolutional Neural Networks) [2]. Ici, nous nous concentrons sur les méthodes sans référence (sans information de la meilleure version d'une image) et sans apprentissage.

Ces dernières années, on a vu apparaître de nouvelles techniques basées sur des méthodes d'apprentissage Les premiers travaux ont basé leur modèle sur les caractéristiques du système visuel humain. [3] introduit le concept de Just Noticeable Blur (JNB). [4] ont utilisé la probabilité cumulative de la détection de flou dans sa métrique. Plus récemment, l'analyse des gradients locaux des contours permet à [5] de développer l'index PSI (Perceptual Sharpness Index). [6] a analysé aussi les gradients au niveau du pixel afin de calculer la distribution des variations maximales locales (MLV). [7] a utilisé les moments discrets de Tchebychef car leurs magnitudes décroît avec le niveau de flou dans l'image. Cependant, la plupart de ces méthodes sont peu rapides et les méthodes robustes existantes sont coûteuses en calcul. Dans les travaux plus récents, [8] utilise la transformée en ondelettes afin de calculer un score qui est basé sur le centile des coefficients d'ondelettes haute fréquence.

2 Détection du flou en temps réel

2.1 Principe du LBP (dérivé de FLASH [1])

L'objectif de l'application qui nous intéresse ici est le contrôle de qualité sur les ouvrages d'art. Ce contrôle de qualité passent en particulier par la détection de fissures. Nous avons développé pour cela un détecteur spécifique, nommé FLASH et présenté dans [1]. Son efficacité est bien meilleure lorsqu'il est appliqué sur des images de bonne qualité, c'est à dire avec

un flou réduit. Lorsque le flou est trop important, il est plus intéressant de reprendre une image, et c'est pour cette raison que la détection de fissures et l'évaluation du flou doivent être réalisées de façon conjointe et idéalement à partir des mêmes descripteurs. Cela permet de simplifier les calculs et d'avoir un gain de temps important pour une application temps réel dans un dispositif avec peu de ressources. Or, la détection des fissures FLASH se base sur un descripteur proche de l'opérateur LBP (motifs binaires locaux) introduit par [9]. Dans notre approche, on exploite 16 pixels autour d'un point central à analyser mais une optimisation algorithmique permet de s'affranchir de l'analyse de tous les pixels. Bien qu'en moyenne 4 pixels soient examinés, nous pouvons pousser l'analyse à 8 points minimum sans perdre énormément en temps de calcul pour une application temps réel. On peut alors obtenir une valeur classique de LBP sur 8 bits et avoir 256 configurations à analyser. Cependant, le rayon d'analyse n'est pas fixé à 1 pixel comme pour le LBP classique qui considère les 8 voisins directs mais va de 2 à 3. On a alors une approximation de l'opérateur LBP avec des points se situant sur le cercle de Bresenham comme sur la figure 1.



FIGURE 1 – Analyse FLASH. (a) Exemple de détection de fissure avec l'opérateur FLASH (b) Motif pour la détection FLASH (c) Points analysés dans FLASH pour la construction de LBP_f

Ce nouvel opérateur sera noté LBP_f . Dans la détection de fissures, nous pouvons également calculer dans une même analyse LBP_f . Le lien entre les deux opérateurs est important. Comme le montrent les expérimentations, l'algorithme n'est pas robuste au flou. La détection de fissures peut donc être évaluée en fonction de notre mesure de flou dans un processus englobant à la fois l'acquisition et la détection des défauts.

2.2 Notre détecteur

De la même façon que les algorithmes existants, nous exploitons les structures fines de l'image pour savoir si elle est floue. Comme dans [10], nous basons notre mesure sur le fait que une version floutée d'une image déjà floue au départ sera peu différente de l'image floue d'origine, alors que la version floutée d'une image plus nette au départ présentera des différences plus importantes avec la version plus nette de départ. En effet, les transitions nettes sont plus impactées par une opération de floutage que les transitions présentes dans une image déjà floue. Concrètement, le principe consiste à comparer l'image originale avec sa version artificiellement floutée. Cette dernière est obtenue en appliquant un filtre passe-bas (filtre moyenneur ou filtre gaussien sur l'image d'origine).

En pratique, l'histogramme des motifs uniformes LBP sert d'indicateur destiné à évaluer le flou. Nous utiliserons pour caractériser le flou, la somme normalisée d'une partie de cet histogramme. Cet histogramme est constitué de 10 bins, 0 à 8 pour les motifs uniformes et le bin 9 pour le nombre de motifs non-uniformes. Comme décrit dans [9], chaque bin *i* représente le nombre de motifs correspondant à sa valeur $LBP^{riu2} = i$, comptabilisés sur l'ensemble de l'image. Par exemple, le bin 0 est le nombre de motif ayant une valeur de LBP égale à 0. Le motif 0 n'est pas considéré car une image nette peut avoir des zones complètement homogènes. Dans [11], les auteurs montrent que le profil de l'histogramme des LBP change significativement pour les valeurs entre 6 et 9. Nous avons défini notre mesure *s* comme suit :

$$s = \frac{|sLBP - sLBP_{blurred}|}{\#(LBP_f^{riu2} \neq 0)} \quad \text{avec} \quad sLBP = \sum (LBP_f^{riu2} \geq 6) \tag{1}$$

Le score ne dépend pas d'une zone spécifique comme les contours. Le motif '00000000' n'est pas considéré empêchant toute prise en compte de grandes zones homogènes comme un ciel bleu ou un mur peint lisse. La difficulté dans les images peu texturées est de trouver de petits indices qui permettent de déterminer la qualité de l'image. Les motifs non-uniformes ont une large contribution. En effet, il montre la présence d'un grand nombre de transitions dans une région locale. Plus la valeur de notre score est élevée, plus l'image est nette.

2.3 Évaluation préliminaire

Temps de calcul. Le temps de calcul est primordial si l'on veut assurer une acquisition optimale en qualité en temps réel. Le tableau 1 montre un récapitulatif des tests effectués avec les algorithmes disponibles. Notre mesure présente des résultats très compétitifs en temps de calcul. Ces temps sont calculés à l'aide de MATLAB sur l'ensemble de notre jeu de données.

L'algorithme ARISMC [12] a pris plus de 2 jours de traitement. La méthode MLV présente le résultat le plus faible en terme de temps de traitement après ARISMC. De même l'algorithme PSI [5] obtient une bonne performance, en temps de calcul, en comparaison avec notre mesure. Notre méthode possède un avantage certain : c'est la possibilité d'effectuer le calcul uniquement sur la version floutée de l'image afin d'obtenir l'indice de flou. Ainsi, nous pouvons approximativement diviser le temps de traitement par un facteur 2.

Méthodes	Notre mesure	BISHARP [8]	MLV [6]	ARISMC[12]	PSI [5]	BRISQUE [13]
Temps (s)	536	1287	1597	+40000	632	1141

TABLE 1 – Temps de calcul pour traiter l'ensemble de notre base de données d'images de surface de béton. Les temps donnés sont en secondes et correspondent à une exécution d'un code Matlab. En pratique, l'implémentation embarquée permet le temps réel.

3 Perspectives

Nous envisageons de mettre en corrélation le seuil critique de détection de défauts type fissure avec notre mesure de qualité d'une image par rapport au flou. Plusieurs scénarios sont à élaborer et nous devons évaluer la corrélation la robustesse des images issues de scènes différentes. L'alliance de la détection et d'un indicateur de fiabilité de la détection des défauts est un avantage durant l'acquisition.

Nous étudions également l'utilisation des motifs binaires locaux et ses dérivés afin d'ajouter de nouveaux types de distorsions des images dans notre algorithme de contrôle qualité des images.

Références

- [1] Yannick Faula, Stéphane Bres, and Véronique Eglin, "FLASH : A New Key Structure Extraction used for Line or Crack Detection :," in *Proceedings of the 13th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications*, Funchal, Madeira, Portugal, 2018, pp. 446–452, SCITEPRESS - Science and Technology Publications.
- [2] Y. Li, Z. Wang, G. Dai, S. Wu, S. Yu, and Y. Xie, "Evaluation of realistic blurring image quality by using a shallow convolutional neural network," in 2017 IEEE International Conference on Information and Automation (ICIA), July 2017, pp. 853–857.
- [3] R. Ferzli and L. J. Karam, "A No-Reference Objective Image Sharpness Metric Based on the Notion of Just Noticeable Blur (JNB)," *IEEE Transactions on Image Processing*, vol. 18, no. 4, pp. 717–728, Apr. 2009.
- [4] N. D. Narvekar and L. J. Karam, "A no-reference perceptual image sharpness metric based on a cumulative probability of blur detection," in 2009 International Workshop on Quality of Multimedia Experience, July 2009, pp. 87–91.
- [5] C. Feichtenhofer, H. Fassold, and P. Schallauer, "A Perceptual Image Sharpness Metric Based on Local Edge Gradient Analysis," *IEEE Signal Processing Letters*, vol. 20, no. 4, pp. 379–382, Apr. 2013.
- [6] K. Bahrami and A. C. Kot, "A Fast Approach for No-Reference Image Sharpness Assessment Based on Maximum Local Variation," *IEEE Signal Processing Letters*, vol. 21, no. 6, pp. 751–755, June 2014.
- [7] L. Li, W. Lin, X. Wang, G. Yang, K. Bahrami, and A. C. Kot, "No-Reference Image Blur Assessment Based on Discrete Orthogonal Moments," *IEEE Transactions on Cybernetics*, vol. 46, no. 1, pp. 39–50, Jan. 2016.
- [8] Goran Gvozden, Sonja Grgic, and Mislav Grgic, "Blind image sharpness assessment based on local contrast map statistics," *Journal of Visual Communication and Image Representation*, vol. 50, pp. 145–158, Jan. 2018.
- [9] T. Ojala, M. Pietikainen, and T. Maenpaa, "Multiresolution gray-scale and rotation invariant texture classification with local binary patterns," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, no. 7, pp. 971–987, July 2002.
- [10] Till Sieberth, Rene Wackrow, and Jim H. Chandler, "Automatic detection of blurred images in UAV image sets," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 122, pp. 1–16, Dec. 2016.
- [11] X. Yi and M. Eramian, "LBP-Based Segmentation of Defocus Blur," *IEEE Transactions on Image Processing*, vol. 25, no. 4, pp. 1626–1638, Apr. 2016.
- [12] K. Gu, G. Zhai, W. Lin, X. Yang, and W. Zhang, "No-Reference Image Sharpness Assessment in Autoregressive Parameter Space," *IEEE Transactions on Image Processing*, vol. 24, no. 10, pp. 3218–3231, Oct. 2015.
- [13] A. Mittal, A. K. Moorthy, and A. C. Bovik, "No-Reference Image Quality Assessment in the Spatial Domain," *IEEE Transactions on Image Processing*, vol. 21, no. 12, pp. 4695–4708, Dec. 2012.