

Segmentation hiérarchique faiblement supervisée

Amin FEHRI, Santiago VELASCO-FORERO, Fernand MEYER

Mines ParisTech, PSL Research University, Centre de Morphologie Mathématique,
35 rue Saint-Honoré, 77305 Fontainebleau Cedex, France

{amin.fehri, santiago.velasco, fernand.meyer@mines-paristech.fr}

Résumé – La segmentation d’images est le processus visant à partitionner une image en régions d’intérêt par rapport à des critères donnés. La segmentation hiérarchique s’est révélée être une approche majeure en ce sens, car elle favorise l’émergence des régions importantes à différentes échelles. D’autre part, de nombreuses méthodes nous permettent aujourd’hui d’avoir une information a priori sur la position des structures d’intérêt dans les images. Dans cet article, nous proposons un algorithme versatile de segmentation hiérarchique qui permet de prendre en compte une telle information a priori pour donner une segmentation hiérarchique qui met l’accent sur les régions ou contours d’intérêt tout en préservant les structures importantes dans l’image. Une application au problème de segmentation faiblement supervisée est présentée.

Abstract – Image segmentation is the process of partitioning an image into a set of meaningful regions according to some criteria. Hierarchical segmentation has emerged as a major trend in this regard as it favors the emergence of important regions at different scales. On the other hand, many methods allow us to have prior information on the position of structures of interest in the images. In this paper, we present a versatile hierarchical segmentation method that takes into account any prior spatial information and outputs a hierarchical segmentation that emphasizes the contours or regions of interest while preserving the important structures in the image. An application of this method to the weakly-supervised segmentation problem is presented.

1 Introduction

Dans cet article, nous proposons une méthode permettant de tirer parti de toute information spatiale préalablement obtenue sur une image pour obtenir une segmentation hiérarchique de cette image qui met en exergue ses régions d’intérêt. Ceci nous permet d’avoir plus de détails dans ces régions tout en conservant les informations structurelles fortes de l’image.

Les potentielles applications sont nombreuses. Cela permettrait par exemple, lorsqu’on a une capacité de stockage limitée (e.g. pour de très larges images), de conserver les détails de l’image dans les régions d’intérêt en priorité. De nombreux cas étant abordés dans un article associé [5], nous illustrerons dans la suite cette méthode uniquement par son application à un problème de segmentation faiblement supervisé. Une vue d’ensemble est proposée en figure 1.

Il a été montré que la segmentation est de façon inhérente un problème multi-échelles. C’est pourquoi la segmentation hiérarchique est devenue une tendance majeure ces dernières années, et qu’elle a donné la plupart des algorithmes de l’état de l’art actuel en segmentation. L’idée est de pas renvoyer uniquement une partition des pixels de l’image mais une structure multi-échelles qui rende compte des objets d’intérêt à toutes les échelles. Nous présentons ici un algorithme de segmentation hiérarchique qui se concentre sur certaines zones prédéterminées de l’image. L’aspect hiérarchique permet alors de choisir de façon simple le niveau de détails désiré en fonction de l’application. De plus, cet algorithme est très versatile car l’information spatiale a priori qu’il prend en entrée peut être obtenu par n’importe laquelle des nombreuses approches de vision par ordinateur existantes pour localiser les objets. De ce point de vue,

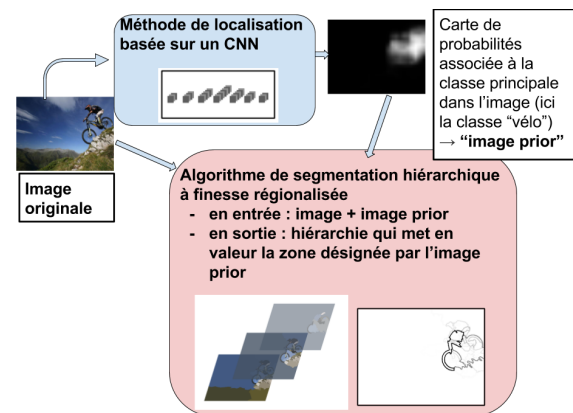


FIGURE 1 – Vue d’ensemble de la méthode proposée.

ce travail s’inscrit dans le cadre des recherches visant à incorporer un savoir pré-existant sur l’image pour la segmentation.

Dans la section 2, nous expliquerons comment construire et utiliser un algorithme de segmentation hiérarchique sur graphes. Puis nous préciserons dans la section 3 comment utiliser une information spatiale a priori sur un objet d’intérêt pour obtenir une hiérarchie à finesse régionalisée. Enfin, un exemple d’application à la segmentation faiblement supervisée sera présenté dans la partie 4 et des conclusions seront tirées dans la partie 5.

2 Hiérarchies et partitions

2.1 Segmentation hiérarchique fondée sur des graphes

Obtenir de façon directe une segmentation adaptée à partir d’une image est très difficile. C’est pourquoi on fait souvent

appel à des hiérarchies pour organiser et proposer des contours intéressants en les valuant. Dans cette section, nous rappelons au lecteur comment construire et utiliser des outils de segmentation hiérarchique sur graphes.

On suppose que, pour chaque image, une partition fine est obtenue par une segmentation initiale et qu'elle contient tous les contours qui font sens dans l'image. On définit alors une mesure de dissimilarité entre tuiles adjacentes de cette partition fine. On peut alors voir l'image comme un graphe, le *graphe d'adjacence de régions* (GAR), dans lequel chaque noeud représente une tuile de la partition fine; une arête relie deux noeuds si les régions correspondantes sont voisines dans la partition fine; le poids de chaque arête est égal à la dissimilarité entre ces régions. Travailler sur le GAR est beaucoup plus efficace que de travailler sur l'image, car il y a beaucoup moins de noeuds dans le GAR qu'il n'y a de pixels dans l'image.

De façon formelle, on note ce graphe $\mathcal{G} = (\mathcal{V}, \mathcal{E}, \mathbf{W})$, où \mathcal{V} correspond au domaine de l'image ou à l'ensemble des pixels ou régions fines, $\mathcal{E} \subset \mathcal{V} \times \mathcal{V}$ est l'ensemble des arêtes reliant les régions voisines, $\mathbf{W} : \mathcal{E} \rightarrow \mathbb{R}^+$ est la valeur de dissimilarité généralement basée sur une mesure du gradient local (ou couleur ou texture), par exemple $\mathbf{W}(i, j) \propto |\mathbf{I}(v_i) - \mathbf{I}(v_j)|$ avec $\mathbf{I} : \mathcal{V} \rightarrow \mathbb{R}$ représentant l'intensité de l'image.

L'arête reliant les noeuds p et q est notée e_{pq} . Un *chemin* est une suite de noeuds et d'arêtes : par exemple, un chemin liant les noeuds p et q est l'ensemble $\{p, e_{pt}, t, e_{ts}, s\}$. Un *graphe connecté* est un sous-graphe dans lequel toute paire de noeuds est connectée par un chemin. Un *cycle* est un chemin dont les extrémités coïncident. Un *arbre* est un graphe connecté sans cycle. Un *arbre couvrant* est un arbre contenant tous les noeuds. Un *arbre de poids minimum* (APM) $\mathcal{APM}(\mathcal{G})$ d'un graphe \mathcal{G} est un arbre couvrant avec le poids minimal possible, obtenu par exemple en utilisant l'algorithme de Boruvka (le poids d'un arbre étant égal à la somme des poids de ses arêtes). Une *forêt* est un ensemble d'arbres.

Une *partition* π d'un ensemble \mathcal{V} est une collection de sous-ensembles de \mathcal{V} , tels que l'ensemble \mathcal{V} soit l'union disjointe des sous-ensembles dans la partition, i.e. $\pi = \{R_1, R_2, \dots, R_k\}$ tel que : $\forall i, R_i \subseteq \mathcal{V} ; \forall i \neq j, R_i \cap R_j = \emptyset ; \bigcup_i R_i = \mathcal{V}$.

Couper toutes les arêtes de $\mathcal{APM}(\mathcal{G})$ ayant une valuation supérieure à un seuil λ mène à une forêt de poids minimum (FPM) $\mathcal{F}(\mathcal{G})$, i.e. à une partition du graphe. La partition obtenue est alors la même que celle qui aurait été obtenue en coupant les arêtes de valuations supérieures à λ directement dans \mathcal{G} . Puisque travailler sur un $\mathcal{APM}(\mathcal{G})$ est moins coûteux et fournit des résultats similaires pour la segmentation, nous travaillons dans la suite uniquement avec un $\mathcal{APM}(\mathcal{G})$.

Couper les arêtes par valuation décroissante mène donc à une *hiérarchie de partitions indexée* (\mathcal{H}, λ) , avec \mathcal{H} une *hiérarchie de partitions* i.e. une chaîne de partitions imbriquées $\mathcal{H} = \{\pi_0, \pi_1, \dots, \pi_n | \forall j, k, 0 \leq j \leq k \leq n \Rightarrow \pi_j \sqsubseteq \pi_k\}$, avec π_n la partition à une seule région et π_0 la partition fine de l'image, et $\lambda : \mathcal{H} \rightarrow \mathbb{R}^+$ étant un indice de stratification vérifiant $\lambda(\pi) < \lambda(\pi')$ pour deux partitions imbriquées $\pi \subset \pi'$. Cette application croissante nous permet d'attribuer à

chaque tronçon de contour le niveau de la hiérarchie pour lequel il disparaît : c'est la *saillance* du contour et on considère que plus la saillance est grande, plus le contour est fort. Pour une hiérarchie donnée, l'image dans laquelle chaque contour prend comme valeur sa saillance est appelée *Carte de Contours Ultramétriques* (CCU)[2]. Représenter une hiérarchie par sa CCU est une façon simple d'avoir une idée de son effet car seuiliser une CCU aboutit toujours à un ensemble de courbes fermées et donc à une partition. Dans cet article, pour une meilleure visibilité, nous représentons les CCU avec un contraste inversé.

Il y a plusieurs façons d'obtenir une segmentation à partir d'une hiérarchie : (i) de façon simple en seuillant les saillances les plus hautes; (ii) en marquant certains noeuds comme importants et en calculant alors une partition conformément à ces marqueurs, connu sous le nom de *segmentation avec marqueurs*; (iii) en recherchant la segmentation minimisant une fonctionnelle énergétique définie sur le graphe.

Or, la qualité des segmentations obtenues dépend grandement de la dissimilarité que nous utilisons, et en changer peut mener à des segmentations plus pertinentes. La ligne de partage des eaux stochastique est un des outils possibles nous permettant d'attribuer aux contours des poids prenant en compte une information plus riche qu'une simple dissimilarité locale, de telle sorte que des méthodes plus simples soient suffisantes pour en extraire des segmentations intéressantes.

2.2 Hiérarchies de Ligne de Partage des Eaux Stochastique (LPES)

La ligne de partage des eaux stochastique (LPES), introduite dans [1] par simulation et étendue à une approche par graphes dans [6], est un outil versatile pour construire des hiérarchies. L'idée originale est d'opérer à de multiples reprises une segmentation avec marqueurs avec des marqueurs aléatoires puis de valuer chaque arête de l' \mathcal{APM} par sa fréquence d'apparition dans les segmentations résultantes.

En effet, en distribuant des marqueurs sur le GAR \mathcal{G} , on peut construire une segmentation comme une FPM $\mathcal{F}(\mathcal{G})$ dans laquelle chaque arbre prend racine dans un noeud marqué. On constate que la segmentation avec marqueurs est possible directement sur l' \mathcal{APM} : on doit alors couper, pour chaque paire de marqueurs, l'arête la plus haute sur le chemin les reliant. De plus, il y a un domaine de variation dans lequel chaque marqueur peut se déplacer tout en aboutissant à la même segmentation. Plus de détails sont fournis dans l'article associé [5].

Considérons sur l' \mathcal{APM} une arête e_{st} de poids ω_{st} et calculons sa probabilité d'être coupée. Nous coupons toutes les arêtes de l' \mathcal{APM} ayant un poids supérieur ou égal à ω_{st} , ce qui aboutit à deux arbres T_s et T_t de racines s et t . Si au moins un marqueur tombe dans le domaine R_s des noeuds de T_s et au moins un marqueur tombe dans le domaine R_t des noeuds de T_t , alors e_{st} sera coupé dans la segmentation finale.

Si l'on note $\mu(R)$ le nombre de marqueurs aléatoires tombant dans la région R , nous voulons attribuer à e_{st} la valeur de probabilité suivante :

$$\begin{aligned}
\tilde{\omega}_{st} &= \mathbb{P}[(\mu(\mathbf{R}_s) \geq 1) \wedge (\mu(\mathbf{R}_t) \geq 1)] \\
&= 1 - \mathbb{P}[(\mu(\mathbf{R}_s) = 0) \vee (\mu(\mathbf{R}_t) = 0)] \\
&= 1 - \mathbb{P}(\mu(\mathbf{R}_s) = 0) - \mathbb{P}(\mu(\mathbf{R}_t) = 0) \\
&\quad + \mathbb{P}(\mu(\mathbf{R}_s \cup \mathbf{R}_t) = 0)
\end{aligned} \tag{1}$$

Si les marqueurs sont distribués suivant une distribution de Poisson, alors pour une région R :

$$\mathbb{P}(\mu(R) = 0) = \exp^{-\Lambda(R)}, \tag{2}$$

avec $\Lambda(R)$ l'espérance du nombre de marqueurs tombant dans R . La probabilité devient donc :

$$\tilde{\omega}_{st} = 1 - \exp^{-\Lambda(\mathbf{R}_s)} - \exp^{-\Lambda(\mathbf{R}_t)} + \exp^{-\Lambda(\mathbf{R}_s \cup \mathbf{R}_t)} \tag{3}$$

Avec, lorsque la distribution de Poisson a une densité uniforme λ :

$$\Lambda(R) = \text{aire}(R)\lambda, \tag{4}$$

Et lorsque la distribution de Poisson a une densité non-uniforme λ :

$$\Lambda(R) = \int_{(x,y) \in R} \lambda(x,y) dx dy \tag{5}$$

La sortie de l'algorithme de LPES dépend donc de l' \mathcal{APM} de départ (structure et valuations des arêtes) et de la loi de probabilité gouvernant la distribution des marqueurs. De plus, les hiérarchies de LPES peuvent être chaînées puisque les nouvelles valuations des contours correspondent à des nouvelles valeurs pour l' \mathcal{APM} et que l'on peut alors réitérer le processus décrit précédemment sur cet \mathcal{APM} . Ceci amène un large espace d'exploration qui peut être utilisé dans une chaîne de segmentation [4].

Du fait de sa versatilité et de ses bonnes performances, la LPES représente un bon algorithme de départ à modifier pour y injecter une information spatiale a priori. En effet, lorsque l'on a une telle information sur l'image, est-il possible de l'utiliser pour avoir plus de détails dans certaines parties de l'image plutôt que dans d'autres ? Cela permettrait de simplifier et d'améliorer des traitements ultérieurs en concentrant les efforts dans les zones d'intérêt.

3 Hiérarchies avec information a priori

3.1 Hiérarchie à finesse régionalisée (HFR)

Dans la LPES originale, une distribution uniforme de marqueurs est utilisée (quelque soit leur taille ou leur forme). Pour avoir des contours plus forts dans une région spécifique de l'image, on propose d'adapter le modèle pour que plus de marqueurs soient distribués dans cette région.

Soit E un objet d'une classe d'intérêt, par exemple $E = \text{"vélo"}$, et \mathbf{I} l'image étudiée. On note θ_E la densité de probabilités associée à E , définie sur le domaine D de \mathbf{I} et obtenue par ailleurs. On note $CP(\mathbf{I}, \theta_E)$ la carte de probabilités associée, dans laquelle chaque pixel $p(x,y)$ de \mathbf{I} prend comme valeur $\theta_E(x,y)$ sa probabilité de faire partie de E . Etant donnée une telle information sur la position d'un objet d'intérêt dans une image, on obtient une hiérarchie de segmentations focalisée sur cette région en modulant la distribution des marqueurs.

Si λ est une densité (uniforme ou non) définie sur D pour distribuer des marqueurs, on choisit $\theta_E \lambda$ comme nouvelle densité, favorisant ainsi l'émergence de contours parmi les régions

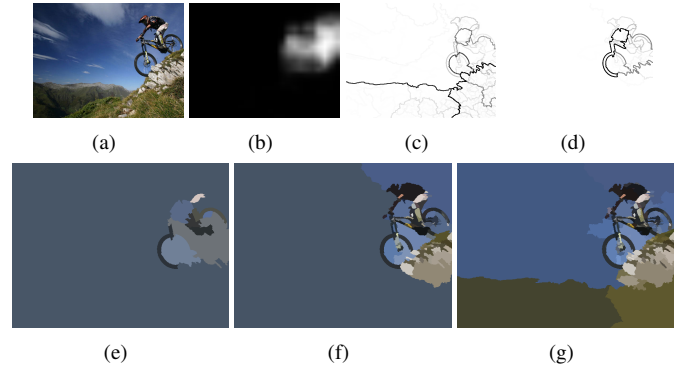


FIGURE 2 – HFR focalisée sur la classe principale (“vélo”) avec une CP issue d’une méthode basée sur des CNN. (a) Image, (b) Heatmap de la classe principale, (c)(d)CCU de la LPES volumique et de la HFR avec information prior. (e)-(g) Segmentations obtenues avec la HFR - 10,100,200 régions.

d'intérêt. Si on considère une région R de l'image, le nombre moyen de marqueurs tombant dans R est alors :

$$\Lambda_E(R) = \int_{(x,y) \in R} \theta_E(x,y)\lambda(x,y) dx dy \tag{6}$$

De plus, cette approche peut aisément être étendue au cas où l'on veut tirer parti d'informations provenant de sources multiples. En effet, si θ_{E_1} et θ_{E_2} sont des densités de probabilités associées aux événements E_1 et E_2 , on peut combiner ces deux sources en utilisant comme nouvelle densité $(\theta_{E_1} + \theta_{E_2})\lambda$.

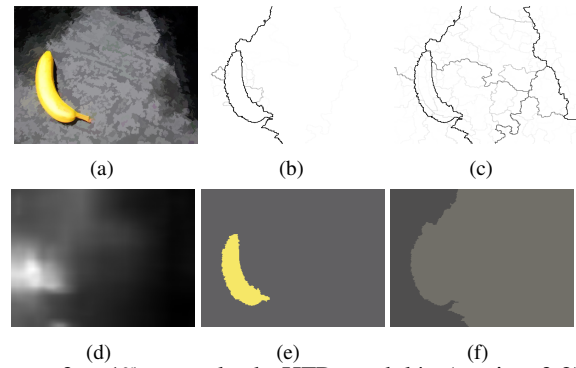


FIGURE 3 – 1^{er} exemple de HFR modulée (section 3.2). (a) Image, (b)(c) CCU de la HFR et de la LPES volumique, (d) Heatmap de la classe principale, (e)(f) Segmentations avec 2 régions pour la HFR - à gauche - et la LPES volumique - à droite.

3.2 Moduler la HFR en fonction des couples de régions considérées

Si on veut favoriser l'émergence de certains contours, on peut moduler la densité de marqueurs dans chaque région en prenant en compte la force des contours les séparant mais aussi leur position relative dans chacune de ces régions.

Par exemple, pour tester la force du gradient séparant les deux régions R_s et R_t , on peut localement distribuer des marqueurs suivant une distribution $\chi(R_s, R_t)\lambda$, avec $\chi(R_s, R_t) = \omega_{st}$. Cela correspond au cas classique de la LPES volumique,

qui permet d’obtenir une hiérarchie prenant en compte à la fois les surfaces des régions et les contrastes entre elles.

On peut utiliser toute information spatiale a priori de façon similaire. En effet, en l’utilisant pour influencer la sortie de l’algorithme de segmentation, on peut vouloir choisir l’information pertinente à mettre en valeur dans les segmentations résultantes : le premier-plan, l’arrière-plan ou les zones de transition entre les deux.

Par exemple, avoir plus de détails dans les régions de transition entre le premier-plan et l’arrière-plan nous permet d’avoir plus de précisions là où la limite entre les deux est justement floue. Considérons ce cas. On définit pour chaque couple de régions (R_s, R_t) un $\chi(R_s, R_t)$ pertinent. On veut alors que $\chi(R_s, R_t)$ soit petit si R_s et R_t sont au premier-plan ou à l’arrière-plan, et grand si R_s est à l’arrière-plan et R_t au premier plan (et vice-versa). On utilise :

$$\begin{cases} \tilde{\lambda} = \chi\lambda \\ \chi(R_s, R_t) = \frac{\max(m(R_s), m(R_t))(1 - \min(m(R_s), m(R_t)))}{0.01 + \sigma(R_s)\sigma(R_t)}, \end{cases} \quad (7)$$

$m(R)$ (resp. $\sigma(R)$) étant la moyenne normalisée (resp. l’écart-type normalisé) des valeurs des pixels dans la région R de $CP(\mathbf{I})$. Ainsi le nombre de marqueurs distribués sera plus grand lorsque le contraste entre les régions adjacentes sera grand (terme au numérateur) et lorsque ces régions seront cohérentes (terme au dénominateur).

4 Segmentation hiérarchique faiblement supervisée

Il existe aujourd’hui de nombreux algorithmes permettant d’obtenir d’obtenir une localisation grossière de l’objet principal dans une image. On s’inspire de celle décrite dans [7] pour ce faire. En utilisant le classifieur de type réseau de neurones convolutifs (CNN) VGG19 [8] entraîné sur la base ImageNet [3], on commence par déterminer quelle est la classe principale dans l’image. Ce CNN prend comme entrée uniquement des images de taille 224×224 pixels. Une fois la classe de l’image déterminée, on peut alors, en changeant l’échelle de l’image d’un facteur $s \in \{0.5, 0.7, 1.0, 1.4, 2.0, 2.8\}$, calculer pour des sous-fenêtres de taille 224×224 de l’image, la probabilité d’apparition de la classe principale. On obtient ainsi une carte de probabilités de la classe principale pour chaque échelle. Par *max-pooling* (“mise en commun en prenant le max”), on garde en mémoire le résultat de l’échelle pour laquelle la probabilité est la plus grande. La *heatmap* (“carte de chaleur”) ainsi générée prend en chaque pixel pour valeur la probabilité que ce pixel appartienne à la classe d’intérêt. Elle peut alors être utilisée en entrée de notre algorithme. De cette façon, nous avons à notre disposition un outil pour se focaliser automatiquement sur la classe principale de chaque scène.

Au final, la combinaison de cette méthode et de la HFR est très intéressante puisqu’elle permet tout à la fois de classifier l’image (i.e. d’en déterminer la classe principale), d’y localiser les éléments de la classe principale et d’obtenir une hiérarchie de segmentations focalisée sur ces éléments, i.e. une hiérarchie

où les contours appartenant à ces éléments sont mis en exergue. Des résultats visuels sont présentés dans les figures 2, 3 et 4.

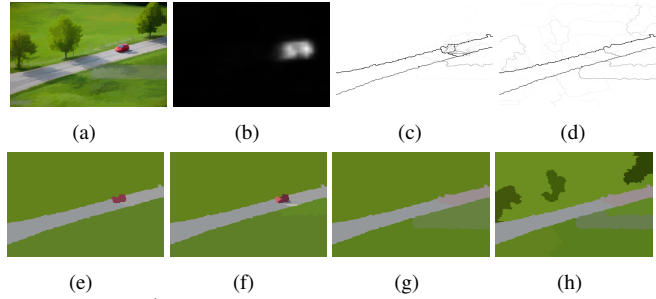


FIGURE 4 – 2^{ème} exemple de HFR modulée (section 3.2). (a) Image, (b) Heatmap de la classe principale, (c)(d) CCU de la HFR et de la LPES volumique (e)-(h) Segmentations de 5 et 10 régions pour la HFR - (e) et (f) - et la LPES volumique - (g) et (h).

5 Conclusion

Dans cet article, nous avons proposé un algorithme novateur et efficace de segmentation hiérarchique mettant l’accent sur des régions d’intérêt d’une image en utilisant une information exogène obtenue sur celle-ci. La grande variété de possibles sources d’information exogène rend notre méthode extrêmement versatile, et nous l’avons ici illustrée sur le problème de segmentation hiérarchique faiblement supervisé. Pour aller plus loin, on pourrait chercher à étendre ce travail à des vidéos.

Références

- [1] Angulo, J., Jeulin, D. : Stochastic watershed segmentation. ISMM 1, 265–276 (2007)
- [2] Arbelaez, P., Maire, M., Fowlkes, C., Malik, J. : Contour detection and hierarchical image segmentation. IEEE PAMI 33(5), 898–916 (2011)
- [3] Deng, J., Dong, W., Socher, R., Li, L.J., Li, K., Fei-Fei, L. : ImageNet : A Large-Scale Hierarchical Image Database. In : CVPR (2009)
- [4] Fehri, A., Velasco-Forero, S., Meyer, F. : Automatic selection of stochastic watershed hierarchies. In : EUSIPCO (August 2016)
- [5] Fehri, A., Velasco-Forero, S., Meyer, F. : Prior-based hierarchical segmentation highlighting structures of interest. In : ISMM (May 2017)
- [6] Meyer, F. : Stochastic watershed hierarchies. In : ICAPR. pp. 1–8 (2015)
- [7] Oquab, M., Bottou, L., Laptev, I., Sivic, J. : Is object localization for free? weakly-supervised learning with convolutional neural networks. In : CVPR (2015)
- [8] Simonyan, K., Zisserman, A. : Very deep convolutional networks for large-scale image recognition. arXiv 1409.1556 (2014)