



HAL
open science

Minimisation du Temps de Réponse moyen d'une Cascade de Détection

Francisco Rodolfo Barbosa-Anda, Cyril Briand, Frédéric Lerasle, Alhayat Ali
Mekonnen

► **To cite this version:**

Francisco Rodolfo Barbosa-Anda, Cyril Briand, Frédéric Lerasle, Alhayat Ali Mekonnen. Minimisation du Temps de Réponse moyen d'une Cascade de Détection. Congrès annuel de la société Française de Recherche Opérationnelle et d'Aide à la Décision (ROADEF), Feb 2016, Compiègne, France. 2p. hal-01765302

HAL Id: hal-01765302

<https://hal.science/hal-01765302v1>

Submitted on 12 Apr 2018

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.

Minimisation du Temps de Réponse moyen d'une Cascade de Détection

F.R. Barbosa-Anda^{1,2}, C. Briand^{1,2}, F. Lerasle^{1,2}, A.A. Mekonnen¹

¹ CNRS, LAAS, 7 avenue du colonel Roche, F-31400 Toulouse, France

² Univ de Toulouse, UPS, LAAS, F-31400 Toulouse, France
{fbarbos, briand, lerasle, aamekonn}@laas.fr

Mots-clés : *Programmation Mathématique, Cascade de Détection, Apprentissage Supervisé.*

1 Introduction

La détection visuelle d'objets est importante dans de nombreux domaines d'application [1]. L'objectif est de déterminer la présence et la position de certains types d'objets dans une image. Les approches existantes réalisent classiquement un échantillonnage de l'image en fenêtres [3], chacune étant analysée pour déterminer si elle contient l'objet recherché (échantillon positif) ou non (échantillon négatif). Cette approche induit un temps de traitement important parce qu'une image génère une grande quantité d'échantillons, surtout négatifs. Pour pallier ce problème, des architectures de classification en cascade ont été proposées [2] dans lesquelles des classificateurs faibles sont utilisés à chaque étage. Un échantillon classifié négatif à un niveau de la cascade est définitivement rejeté (et ne traverse par les niveaux suivants). Les cascades construites avec la méthode AdaBoost comptent parmi les plus performantes [3]. AdaBoost est basé sur un apprentissage supervisé d'un modèle de classification qui utilise une somme pondérée des résultats des classificateurs faibles. À chaque niveau de la cascade est associé un seuil de classification : si le score obtenu par un échantillon à un étage est supérieur ou égale au seuil, ce dernier est classé positif, sinon il est négatif et rejeté. Ces seuils sont choisis afin de respecter un taux de vrais positifs souhaité (TPR). La variante la plus récente de cascade est le modèle *soft-cascade* [4] sur lequel est basé notre travail.

La qualité d'une *soft-cascade* étant principalement mesurée en termes de TPR et de taux de faux positifs (FPR), les méthodes d'apprentissage classiques ne prennent pas en compte explicitement leur temps de réponse, pourtant central dans des applications où l'environnement est fortement dynamique. Dans ce travail, nous envisageons une technique de minimisation du temps de réponse qui consiste à optimiser les seuils de classification utilisés à chaque étage de la cascade. Nous produisons trois contributions : (1) nous prouvons tout d'abord que déterminer des seuils de classification qui minimise le temps de réponse d'une *soft-cascade* est NP-Difficile ; (2) nous proposons ensuite des propriétés de dominance permettant de réduire de façon importante l'espace de recherche de solution ; (3) exploitant ces propriétés, nous introduisons une formulation de programmation linéaire 0 – 1 originale pour résoudre ce problème ; (4) nous validons enfin la méthode proposée de façon expérimentale sur une base de données de la littérature.

2 Définition du problème et principaux résultats

Un *soft-cascade* est composée de L niveaux. À chaque niveau l de la cascade sont associés un poids α_l et un temps de calcul c_l . La cascade fait un apprentissage sur une base composée de N échantillons partitionnés en J échantillon positifs et K échantillons négatifs. Pour chaque niveau l et pour chaque échantillon n , on connaît la réponse $h_{n,l} \in \{0, 1\}$ de la cascade : $h_{n,l} = 1$

indique que l'échantillon $n \in \mathbf{J} \cup \mathbf{K}$ est identifié positif à l'étage l . Un échantillon n est rejeté ou non au niveau l en fonction à son score $S_{n,l} = \sum_{u \leq l} \alpha_u h_{n,u}$: n est classé positif si $S_{n,l} \geq \theta_l$, où θ_l est le seuil de classification choisi au niveau l . Si l'échantillon est classé positif il passe au prochain niveau de la cascade, sinon il est classé négatif. Le temps de réponse de la cascade pour un échantillon n est $\sum_{u \leq l} c_u$ si l'échantillon est classé négatif au niveau l pour la première fois. On s'intéresse à minimiser la somme totale de ces temps de réponse.

Une méthode classique de sélection de $\Theta = \{\theta_1, \dots, \theta_L\}$ est l'algorithme Direct Backward Pruning (DBP) [4]. Cet algorithme choisi au niveau L l'ensemble $\hat{\mathbf{J}}$ d'échantillons positifs ayant le plus haut score $S_{n,L}$ et tel que $|\hat{\mathbf{J}}| \geq TPR$. Il donne ensuite à chaque seuil θ_l la valeur de score maximale qui permet de classer positif tous les échantillons de $\hat{\mathbf{J}}$. remarquons que cette approche ne prend pas en compte le temps de réponse de la cascade ni le taux FPR .

Dans notre travail, nous proposons de trouver le vecteur de seuils Θ qui minimise le temps de réponse total de la cascade pour les échantillons de la base d'apprentissage, le taux TPR étant imposé. Cette base étant supposée représentative de n'importe quelle base inconnue, ce problème est équivalent à minimiser le temps de réponse moyen d'une soft-cascade (MSCRMP). Sur la base d'une réduction du problème **Subset-Sum**, nous établissons que :

Théorème 1. *MSCRMP est NP-difficile.*

Les évolutions possibles des scores des échantillons peuvent être représentées comme une arborescence dans lequel chaque nœud est associé à un couple score-niveau et où chaque arc entre (l, s) et $(l + 1, s')$ indique la possibilité de passer d'un score à un autre pour au moins un échantillon. Sur cette base, nous prouvons les résultats de dominance suivants.

Propriété 1. *L'espace de recherche d'une solution peut être limité aux vecteurs de seuils Θ tels que :*

- à chaque niveau l , $\exists n : \theta_l = S_{n,l}$ et $\theta_l \leq \theta_{l+1}$;
- deux seuils θ_l et θ_{l+1} correspondant aux sommets (l, s) et $(l + 1, s')$ de l'arborescence de score vérifient $s' \leq S_{n,l+1} \forall n : S_{n,l} = s$.

Ces propriétés indiquent qu'il est possible d'assimiler la recherche d'une solution à MSCRMP à la recherche d'un chemin dans une arborescence de seuils. D'autre part, seuls certains de ces chemins sont dominants. Enfin, étant donnée la valeur TPR , il est possible de filtrer davantage l'espace de recherche. Sur cette base, une formulation de programmation linéaire 0 – 1 (BIP) de type flot est proposée. Ce BIP, implémenté sous GUROBI, a été testé sur une base d'apprentissage de taille moyenne de la littérature. Les solutions trouvées ont été comparées avec celles produites par DBP. Les résultats montrent que notre approche optimale génère des vecteurs de seuils ayant un temps de réponse bien meilleur que ceux trouvés par DBP (jusqu'à 20% de gain). Malheureusement, le solveur ne permet pas de traiter des grosses instances comptant plusieurs milliers d'échantillons et plusieurs centaines d'étages, cette taille étant celle des bases d'apprentissage usuelles. Ces résultats justifient la recherche de méthodes d'optimisation plus ad-hoc, capables de résoudre des instances réalistes.

Références

- [1] M.D. Breitenstein, F. Reichlin, B. Leibe, E. Koller-Meier, and L. Van Gool. Online multi-person tracking-by-detection from a single, uncalibrated camera. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 33(9) :1820–1833, 2011.
- [2] P. Dollár, R. Appel, S. Belongie, and P. Perona. Fast feature pyramids for object detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 36(8) :1532–1545, 2014.
- [3] P. A. Viola and M. J. Jones. Robust real-time face detection. *International Journal of Computer Vision*, 57(2) :137–154, 2004.
- [4] Cha Zhang and Paul A Viola. Multiple-instance pruning for learning efficient cascade detectors. In *Advances in Neural Information Processing Systems (NIPS'08)*, pages 1681–1688, 2008.