

IN52 - Traitement et analyse d'images numériques
IN54 - Reconnaissance des formes

RAPPORT DE PROJET

IMAGES AERIENNES



Mathilde Perrot
Romain Begagnon
Emilien Moncan
Remi Ponnelle
Nahil Zamiati

Automne 2018

Sommaire

Introduction	3
Méthodes envisagées.....	4
Deep Learning	4
SVM & HOG	5
Autres techniques	6
Python et OpenCV	6
Travail réalisé	8
Méthode.....	8
Résultats.....	11
Limites	13
Conclusion	15

Introduction

Détecter un objet spécifique sur une image est une chose triviale pour une personne. Toutefois, l'automatisation de cette tâche à l'aide d'un ordinateur s'avère être un véritable défi. Le domaine de la détection d'objet est un domaine qui a subi récemment de nombreuses recherches novatrices avec des résultats défiant toutes les attentes. Cependant, le passage du jeu de test à la vie réelle peut devenir une difficulté qu'il faut surmonter.

Aujourd'hui, de nombreux domaines ont besoin de la détection d'objets à divers niveaux, que ce soit la reconnaissance automatique de pièces en usine ou au sein d'applications plus novatrices comme la voiture automobile. Le traitement et l'analyse d'images numériques est donc de plus en plus important et trouve toujours plus d'applications, diverses et variées.

D'autre part, cette détection et reconnaissance d'objets ou de structures peut s'appliquer à des domaines beaucoup plus particuliers. En effet, l'idée de base de ce projet partait d'un besoin relié à l'écologie ou même à la géopolitique : un article mettait en évidence les intérêts d'une reconnaissance de points d'eau à partir d'images satellites. Une application comme celle-ci permettrait ainsi de reconnaître des sources d'eau dans des zones sensibles, ou tout simplement surveiller l'évolution des fleuves ou des lacs.

De nombreux enjeux peuvent se rapporter à une application de détection à partir d'images satellites, cependant nous avons choisi de spécialiser notre sujet afin de se concentrer sur un seul type de « points d'eau », les piscines. L'intérêt peut aller d'une simple étude statistique jusqu'à un programme permettant de détecter les parcelles contenant une piscine personnelle, et ainsi reconnaître les piscines non déclarées et faisant perdre d'importantes sommes d'argent à l'Etat. Des logiciels proposent de libeller ces images satellites pour comptabiliser tout ceci à la main, mais l'automatisation de la procédure représenterait un réel gain de temps et d'argent.

Nous avons donc pris la décision de travailler sur la détection et la reconnaissance des piscines, présentes souvent au sein de quartiers d'habitation, et nous allons expliquer notre démarche, notre travail et les difficultés rencontrées dans ce rapport.

Méthodes envisagées

N'ayant pas de connaissances particulières dans le domaine, hormis celle acquies lors du semestre à l'aide des cours, nous avons d'un commun accord choisit d'orienter notre travail différemment de notre idée de départ.

En effet, plutôt que de se concentrer sur une seule application, nous avons trouvé intéressant d'implémenter, de tester et de comparer plusieurs techniques. L'objectif était de comparer ces méthodes, d'en tirer diverses conclusions afin de répondre à notre besoin.

Nous avons finalement rencontré plus de difficultés que prévu, et c'est donc après divers tests et développement que nous nous sommes concentrés sur une technique plus fiable et plus réalisable.

Deep Learning

Tout d'abord, lorsque nous avons eu l'idée de se concentrer sur les piscines, la solution la plus optimale nous a semblé être l'utilisation d'un réseau de neurones, afin de combiner rapidité et fiabilité.

N'ayant jamais eu l'occasion de travailler avec un réseau de neurones complexe, une importante phase de recherche à eu lieu. Nous avons trouvé par hasard un projet déjà existant se rapprochant de ce que nous avions en tête, en Python, se basant sur ArcGIS et utilisant Pytorch.



Aperçu du résultat obtenu par l'application utilisant ArcGIS

C'est en essayant d'implémenter ce programme qu'un problème majeur a été mis en évidence. Peu importe la technique mise en œuvre derrière l'entraînement, nous ne disposons pas d'une base de données pour entraîner notre réseau de neurones.

En effet, nous avons essayé d'utiliser le logiciel lié à cette application, ArcGIS Pro, qui permet de labeliser des parcelles contenant des piscines manuellement, afin d'obtenir quelques données, mais sans succès. En effet, pour qu'un réseau de neurones soit efficace, les données d'apprentissage

doivent être importantes (« the more the data, the better the results »). Nous n'avons pas trouvé de données comprenant des labels semblables sur Internet, et reconnaître manuellement 2000 piscines nous aurait demandé beaucoup trop de temps.

Malgré des résultats probablement excellents, nous avons choisi de trouver un autre moyen, ne possédant pas toutes les ressources pour la mise en place d'une solution semblable.

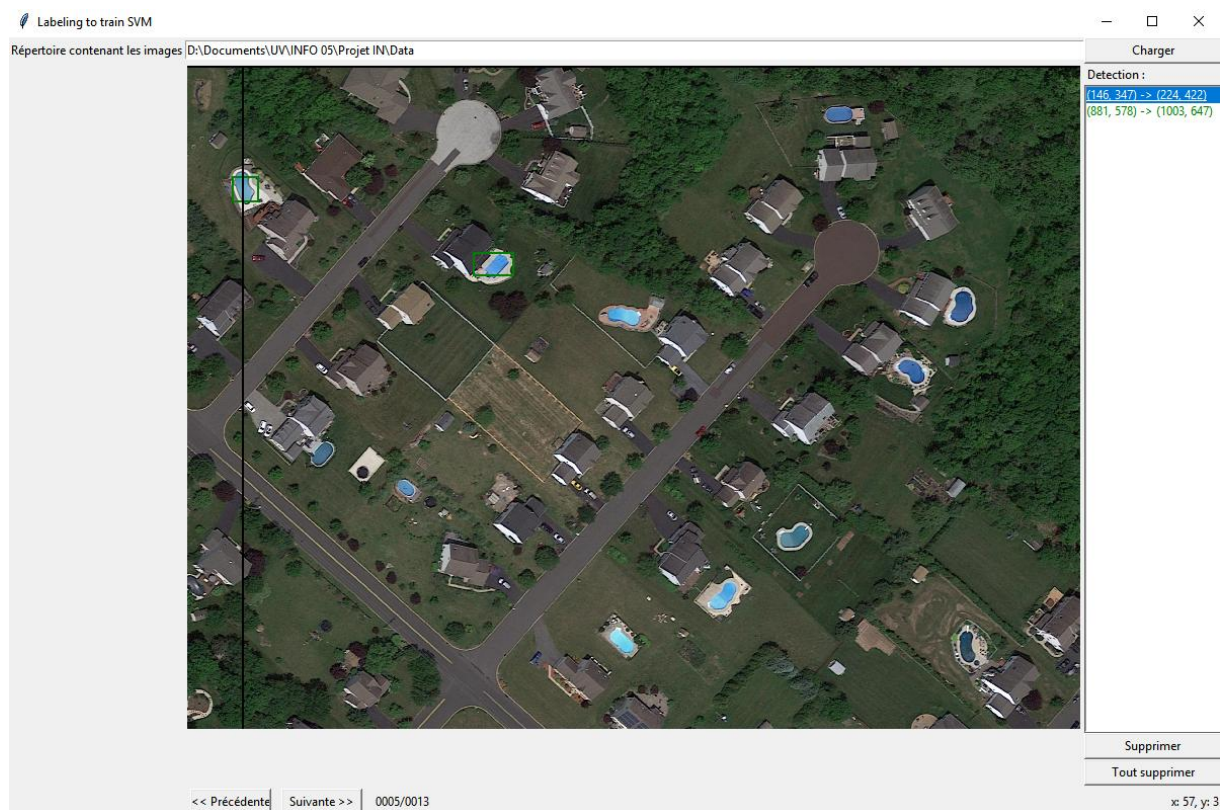
SVM & HOG

En parallèle du travail de recherche sur le Deep Learning, un important travail de recherche a également eu lieu autour de la mise en place d'une SVM (Machine à Vecteurs de Support) couplé à un HOG (Histogramme de gradient orienté).

En effet, cette solution est très souvent utilisée lors de la détection de visages, ou la détection de piétons par exemple. La SVM, faisant office de classifieur, peut ainsi être entraînée à reconnaître des éléments précis à partir de leur histogramme.

Même si le Deep Learning a souvent eu des résultats bien supérieur, l'utilisation de ces deux principes continue de fonctionner avec une bonne fiabilité. Cependant, à la manière d'un réseau de neurones, la SVM doit être entraînée : un classifieur reconnaissant des visages ne reconnaîtra pas les piscines.

Ainsi, un outil permettant de labeliser les images a été créé, afin d'exporter des coordonnées des piscines reconnues manuellement.



Fenêtre de l'outil de labélisation dans le but d'entraîner la SVM

Cependant, en utilisant l'outil afin d'extraire une base de données permettant l'apprentissage, nous nous sommes rendu compte de l'impossibilité de mener l'idée à bien. En effet, sélectionner toutes les piscines manuellement est très fastidieux, et même en obtenant toutes les données sur les quelques images satellites que nous possédons, cela ne sera jamais réellement suffisant pour que la SVM soit assez performante.

Une fois la SVM entraînée, cette technique doit correctement fonctionner, cependant encore une fois les ressources n'étaient pas suffisantes pour mettre en place ce système.

Autres techniques

Nous avons réfléchi alors à d'autres techniques, sans passer par une phase d'apprentissage, évitant l'obligation de réaliser une importante base de test manuellement.

Après d'autres recherches, plusieurs techniques nous ont semblés utilisables, puis très vite écartées. On peut mentionner notamment la classification par motif binaire local en cascade (LBP cascade classification), utilisée dans la détection d'objets ou de visage, mais peu applicable dans notre cas. Nous avons également pensé à utiliser un filtre de Canny, permettant de détecter les contours, mais comment différencier la forme d'un toit et la forme d'une piscine, lorsque tous deux sont rectangulaires. Enfin, après s'être renseigné sur la transformée de Hough pour des détecter des lignes, cette solution a été également écartée, tout simplement dû au fait que les piscines n'ont pas forcément des formes rectangulaires.

Nous sommes donc revenus à une idée plus simple, celle de traiter l'image afin de permettre la détection de piscine. Les résultats probables de cette technique étaient très simples à vérifier, notamment en quelques opérations sous le logiciel Pandore, qui permettaient de voir que jouer sur la couleur des piscines était sans doute la meilleure chose à faire dans notre cas. Le but était donc devenu de travailler sur les histogrammes.

Python et OpenCV

Pour travailler sur les histogrammes des images à traiter, une première solution a été envisagée utilisant Python et OpenCV, une bibliothèque graphique libre spécialisée dans le traitement d'image.

En effectuant un traitement au niveau des couleurs grâce à OpenCV, il est possible d'appliquer un masque à l'image permettant d'isoler toutes les sources de couleur bleu, de teinte assez variée, correspondant à un spectre bien précis. Il est possible de jouer sur ce filtre de couleur en transformant l'image RGB en HSL. Les résultats sont nettement améliorés par cette solution.

La détection n'étant pas parfaite, notamment sur certaines piscines de couleur trop foncés ou de couleur proche du vert, les premières limites de ce type d'application sont apparues, malgré des résultats relativement satisfaisants (peu de détection parasite, dû à la couleur relativement singulière des piscines).



Résultat de l'application sous Python et OpenCV

La mise en place de cette application a permis de bien réaliser qu'il était tout à fait possible de détecter la présence de piscine via un simple traitement d'image, mais il était nécessaire d'effectuer des opérations supplémentaires sur cette image afin d'améliorer la fiabilité et l'efficacité du système.

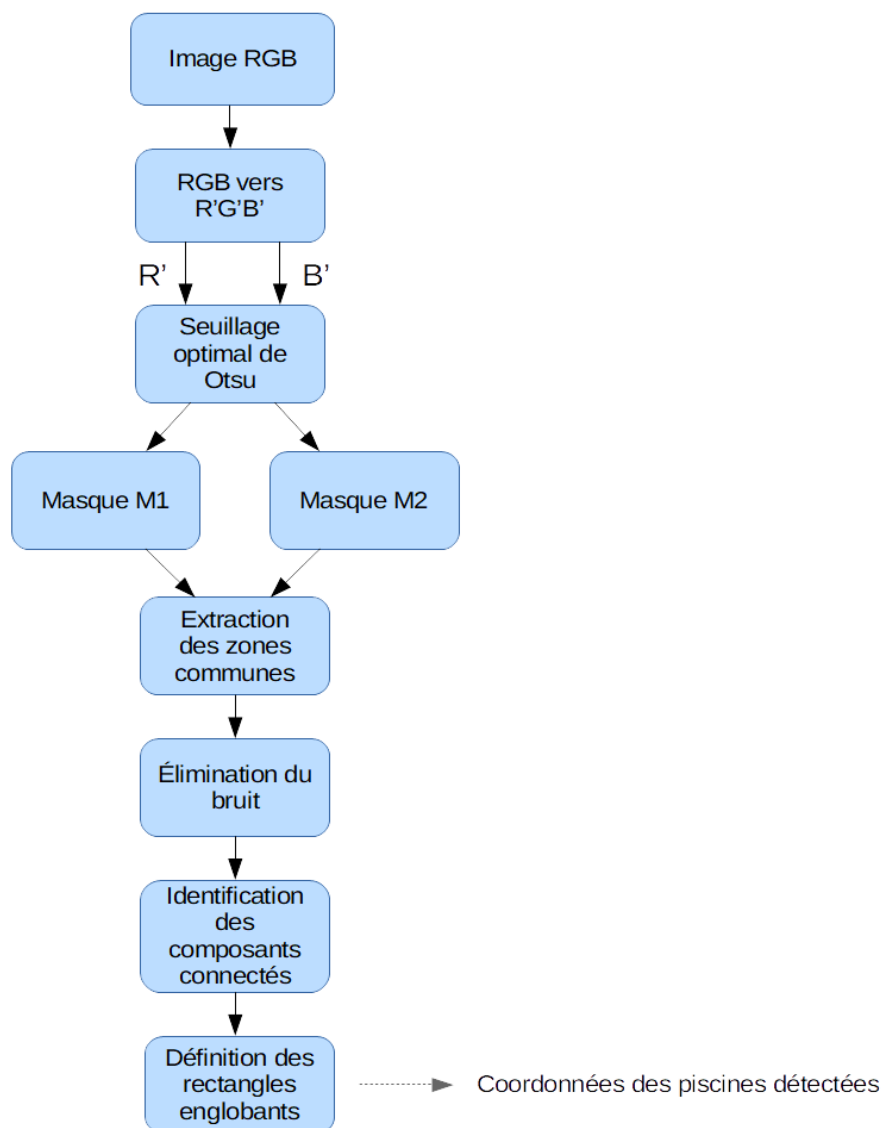
Nous avons décidé par la suite de continuer dans cette optique, mais en utilisant Matlab, avec lequel nous étions plus familiarisé pour le traitement d'image et d'autres outils utiles pour notre application.

Travail réalisé

Cette partie a été implémentée sous Matlab.

Méthode

La méthode utilisée pour cette partie suit le déroulement suivant :



Description de la méthode

Dans un premier temps, l'image est convertie en appliquant aux canaux R, G et B les équations suivantes :

$$R' = \frac{\max - R}{\max - \min}$$

$$G' = \frac{\max - G}{\max - \min}$$

$$B' = \frac{\max - B}{\max - \min}$$

Avec min et max les valeurs minimum et maximum sur l'ensemble RGB.

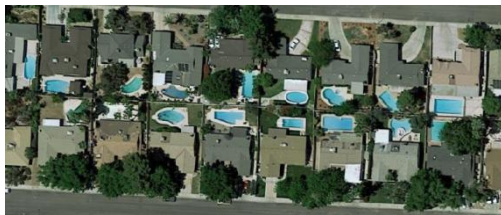


Image RGB



Image R'G'B'

L'image R'G'B' obtenue marque les piscines originalement d'un bleu turquoise en orange très soutenu, tandis que les autres éléments (végétation, bâtiments, etc.) prennent des couleurs plus neutres.

Les canaux R' et B' sont ensuite extraits. Comme le montre la figure suivante, les piscines sont de couleur claire sur le canal R' et elles sont de couleur sombre sur le canal B'. Les autres éléments de l'image, eux, varient peu d'un canal à l'autre.



Canal R'

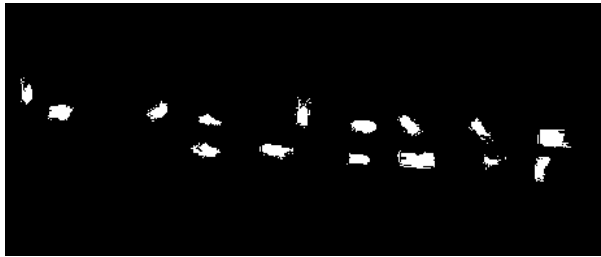


Canal B'

Pour chacune des deux images R' et B' , un seuil est déterminé avec la méthode de seuillage optimal d'Otsu. Les images sont binarisées en utilisant les deux seuils définis. On récupère alors un premier masque regroupant des éléments de couleur claire dans l'image R' , et un second masque regroupant des éléments de couleur sombre dans l'image B' . Les éléments recherchés sont ceux présents dans les deux masques. Pour les extraire et éliminer les éléments non désirés, un produit est appliqué entre les deux masques.

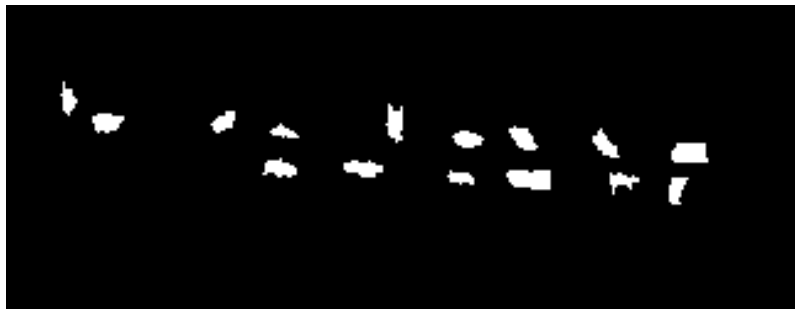


Image originale



Masque obtenu

Une fermeture morphologique est ensuite appliquée sur l'image binaire pour regrouper les pixels susceptibles de faire partie du même élément. De plus, les groupes de pixels trop petits sont considérés comme du bruit et sont retirés du masque. Un groupe est considéré comme trop petit si son nombre de pixels est inférieur à 10% de la moyenne du nombre de pixels de l'ensemble des groupes.



Masque final

Enfin, le rectangle englobant de chaque composant est défini. Chaque rectangle correspond aux coordonnées d'une piscine (x, y, largeur, hauteur) dans l'image aérienne.



Rectangles englobants

Résultats

Plusieurs images ont été marquées manuellement pour permettre la définition de masques de référence pouvant être comparés aux résultats obtenus.



P0061



P0063



P0064



P0133

Images de référence

Image	Nombre de piscines du masque de référence	Bonnes détections	Mauvaises détections	Éléments non détectés	Reconnaissance R	Précision P
P0061.png	118	103	4	15	0.87	0.96
P0063.png	225	199	3	26	0.88	0.99
P0064.png	185	174	3	11	0.94	0.98
P0133.png	20	18	3	2	0.90	0.86

Résultats obtenus

R global : 0.90

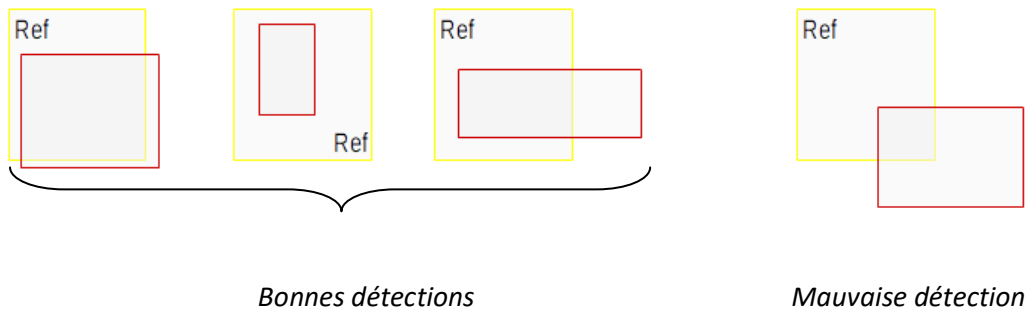
P global : 0.97

Le taux de reconnaissance R et le taux de précision P sont donnés par :

$$R = \frac{\text{nb piscines correctement détectées}}{\text{nb piscines du masque de référence}}$$

$$P = \frac{\text{nb piscines correctement détectées}}{\text{nb total de piscines détectées}}$$

Une piscine est considérée comme correctement détectée si son rectangle englobant est recouvert à au moins 50% par le rectangle englobant d'une piscine du masque de référence.



Comparaison des résultats
En jaune : piscines du masque de référence ; en rouge : piscines détectées

Limites

Les résultats montrés précédemment sont plutôt satisfaisants dans l'ensemble. L'ajout de traitements permet en effet d'améliorer la détection et la reconnaissance. Cependant, il reste quelques cas particuliers expliquant les mauvaises détections ou la non-détection de piscines.

En effet, la détection se base sur la couleur des piscines qui sont d'un bleu soutenu puisqu'elles reflètent la couleur du ciel. En se basant sur ce critère pour la détection, on élimine toutes les piscines qui se différencient par leur couleur. Les piscines dont la couleur tire sur le vert sont

confondues avec le reste du décor, comme la végétation, et ne sont donc pas détectées dans notre application. Les piscines vides ou bâchées ne sont également pas sélectionnées.



Piscines non détectées - Piscine verte à gauche, piscine vide à droite

De plus, les éléments dont la couleur est proche de celle des piscines sont susceptibles d'être intégrés dans la liste des piscines détectées. Ces objets sont peu nombreux sur l'ensemble des images testées car, en plus d'avoir une couleur similaire, ils doivent respecter la taille minimale des piscines recherchées. Cela explique néanmoins qu'il subsiste quelques erreurs de détection (voiture, toit, panneau solaire, etc.).



Mauvaises détections - Panneau solaire à gauche, toiture de bâtiment à droite

Conclusion

Diverses méthodes permettraient théoriquement de mener à bien ce projet. Cependant, face au manque de données, étant très particulières et peu répandus, nous avons du nous rabattre sur une technique de segmentation classique.

L'image subit différentes transformations afin de faire ressortir les piscines. Nous avons pu remarquer que le taux de reconnaissance est de 90% et que le taux de précision avoisine les 97%. Les résultats que nous avons obtenus sont très probants.

Il est toutefois important de noter que cette méthode de détection ne localise pas toutes les piscines et peut se tromper lors de la détection. En effet, des objets dont la couleur se rapproche de celle d'une piscine sont détectés comme tels. De plus, seules les piscines possédant une couleur bleue sont détectées et non celles dont la couleur tire vers le vert de la végétation.

D'autres méthodes seraient peut-être légèrement plus efficaces, pour notamment introduire la reconnaissance des piscines ayant une couleur proche du vert, qui sont liés à un nouvel enjeu, l'hygiène et l'entretien des piscines qui peut être un problème pour la prolifération des moustiques dans certaines région par exemple.