

# Table des matières

Introduction .....	2
I. Etude bibliographique et choix de la méthode .....	3
1) Détection de la plaque.....	3
a. Détection du cadre grâce à la détection de contours.....	3
b. Détection de l'emplacement des caractères .....	4
c. Détection supervisée.....	5
2) Reconnaissance des caractères.....	5
II. Description détaillée de la méthode utilisée .....	8
1) Détection de la plaque.....	8
2) Détection des caractères.....	12
3) Reconnaissance des caractères.....	15
III. Difficultés rencontrées et évaluation des résultats.....	18
Conclusion .....	19
Bibliographie .....	20

# Introduction

La détection et la reconnaissance des plaques d'immatriculation est depuis plusieurs années un enjeu sécuritaire. Elle est en effet très fréquemment utilisée par les forces de l'ordre pour l'identification de voitures volées ou en fuite, ou le trafic de voitures. On la retrouve aussi dans la recherche de plaques d'immatriculation illégales, ou encore l'utilisation de voitures lors d'attaques terroristes ou autres activités frauduleuses.

Du point de vue de la sécurité routière, détecter les plaques d'immatriculation des voitures permet de contrôler le stationnement des véhicules ou la vitesse à laquelle ils roulent, via des radars électroniques, et de contrôler le comportement des conducteurs, via des caméras installées aux feux rouges ou des installations au niveau de barrages de douane par exemple.

Des organismes privés s'intéressent aussi à ces systèmes de détection, notamment pour l'accès à des sites sensibles ou sécurisés.

Le but de ce projet est donc de développer un programme permettant de détecter des plaques d'immatriculation, puis d'en interpréter le contenu, en étudiant différentes méthodes possibles, dans le but d'atteindre le meilleur résultat.

Ce projet est réalisé avec le logiciel Matlab.

# I. Etude bibliographique et choix de la méthode

Le problème posé peut se traiter en deux parties distinctes : d'une part la détection de la plaque, et d'autre part la reconnaissance des caractères. Plusieurs approches existent pour la détection de la plaque. On peut les regrouper en trois catégories :

- Celles qui sont basées sur une détection du cadre de la plaque grâce à une détection des contours.
- Celles qui sont basées sur la détection de l'emplacement des caractères, qui permet de déduire l'emplacement de la plaque.
- Celles qui utilisent une détection supervisée, avec un set de données qui permettent l'apprentissage pour reconnaître la plaque.

## 1) Détection de la plaque

### a. Détection du cadre grâce à la détection de contours

Cette méthode est basée sur une détection de contours en appliquant par exemple un filtre de Canny uniquement sur les lignes horizontales et verticales. Une fermeture permet ensuite de mieux relier les contours entre eux. Pour la partie finale, deux possibilités peuvent être exploitées. D'une part, la détection des caractéristiques structurales particulières met en avant des zones rectangulaires qui potentiellement correspondent à la plaque d'immatriculation. D'autre part, l'étude de l'histogramme cumulé vertical et horizontal permet également de dégager plusieurs zones de contours, dont celle de la plaque (KUMAR, 2018).

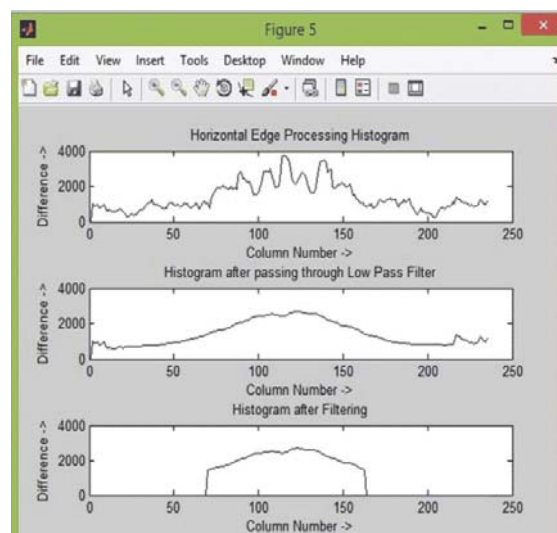


Figure 1 (KUMAR, 2018)

Ameur CHHAYDER et Imene BELHADJ MOHAMED développent plus en détail cette méthode dans leur publication *Système de Reconnaissance Automatique des Plaques Minéralogiques* (CHHAYDER & BELHADJ MOHAMED, 2009).



Figure 2 ( CHHAYDER & BELHADJ MOHAMED, 2009)

Dans le cas de notre problème, cette méthode est peu robuste, en particulier pour les voitures blanches, où il n'y a pas de démarcation entre la plaque et la voiture, et les voitures possédant des motifs (logo, autocollant etc.), ce qui crée des faux positifs.

### **b. Détection de l'emplacement des caractères**

Pour appliquer cette méthode, il faut dans un premier temps faire ressortir les lettres par rapport au reste de l'image. Pour cela, un chapeau haut de forme permet de garder les pics clairs de l'image et donc enlever les petites structures foncées comme les caractères. Une différence entre l'image initiale et l'image après haut de forme permet donc de faire ressortir les petits éléments foncés, ici les caractères de la plaque. Après une binarisation de l'image, une fermeture horizontale de la taille des caractères permet de les relier entre eux et donc de créer un large rectangle blanc au niveau de la plaque. Enfin, des histogrammes cumulés ou une mesure des propriétés de la région pour détecter les rectangles permet de localiser la plaque sur l'image.

Cette méthode a notamment été développée par Régis CLOUARD du projet Panthéon (CLOUARD, 2007).



Figure 3 (CLOUARD, 2007)

Cette méthode est bien plus robuste que la précédente car elle s'affranchit de la couleur du véhicule. Néanmoins, une des voitures de la base de test possède un autocollant d'un numéro de téléphone ce qui crée un faux positif.

### c. Détection supervisée

La détection supervisée est basée sur le fait d'entraîner un réseau de neurones à partir de data set de plaques d'immatriculation. Ce réseau de neurones est ensuite appliqué au set d'images qui doit être testé. Plus le data set utilisé pour l'entraînement est important, plus le pourcentage de détections correctes sera élevé. Or il n'existe pas de set de données de plaques d'immatriculation car il s'agit de données à caractères privées. Comme l'explique toward data science, il est possible de les simuler, en ajoutant de façon aléatoire (taille et position) des plaques des images choisies au hasard, et en ajoutant du bruit (Towards DataScience, 2017).

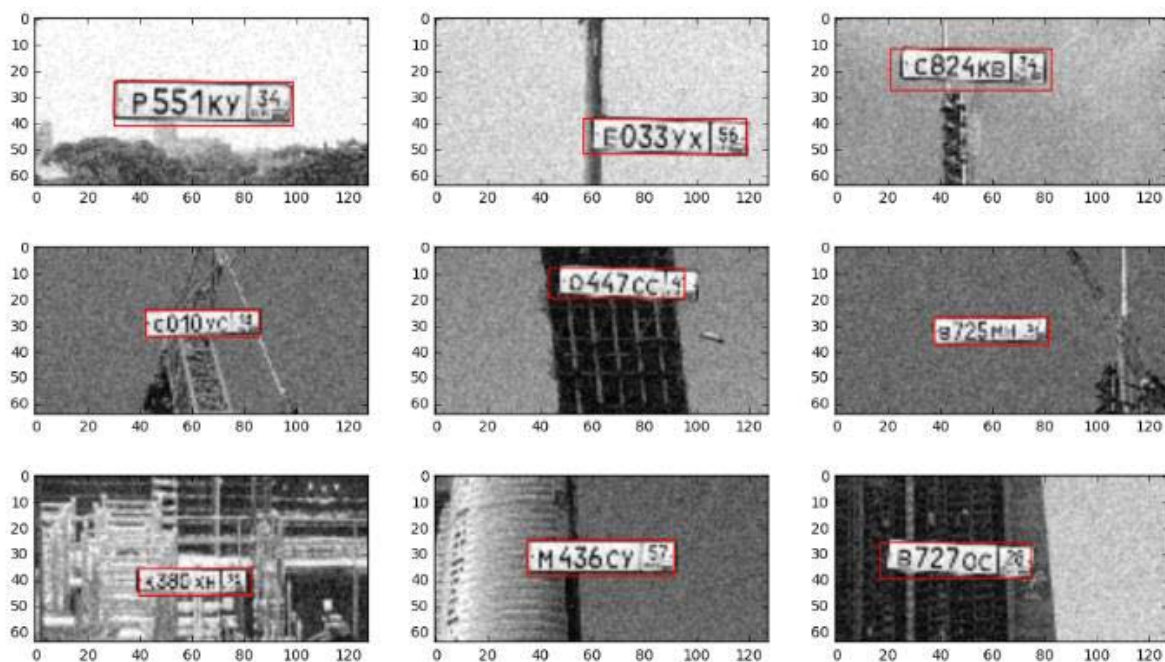


Figure 4 (Towards DataScience, 2017)

Néanmoins, cette méthode est coûteuse en temps et en ressources mémoires et processeurs.

## 2) Reconnaissance des caractères

La deuxième partie du projet consiste à reconnaître les caractères de la plaque. Il existe deux méthodes différentes pour la reconnaissance des caractères :

### a. Template Matching

On utilise des blocs de template matching, dont le but est de donner le meilleur « match » entre le template et l'image d'origine. Ce bloc a deux ports de saisie : le premier est pour l'image de départ, et le second est pour l'image de référence (le template).

La taille du template doit être plus petite que celle de l'image à analyser.

Le bloc de template matching donne la meilleure localisation du template dans l'image d'origine, et renvoie la meilleure valeur correspondante.

Les chiffres sont extraits des plaques d'immatriculation traitées précédemment, et les templates sont issus d'une base de données. L'image d'origine et le template doivent avoir la même orientation et résolution.

Chaque pixel de l'image est comparé à ceux des différents templates par corrélation. Cette corrélation renvoie le taux de corrélation, qui dira si la similitude entre les 2 images ou plus ou moins grande.

Image d'origine :

The image shows a license plate with the text "SUND 045" in a serif font. The characters are black on a light background. The plate is rectangular and has a thin border.

*Figure 5 (Gupta, Purohit, & Rathore, 2014)*

Template :

The image shows a template for license plate extraction. It consists of a grid of characters arranged in five rows. The first row contains digits 0 through 5. The second row contains digits 6 through 9. The third row contains letters A through F. The fourth row contains letters G through M. The fifth row contains letters N through Z. Each character is enclosed in a small square box, and the boxes are arranged in a grid that matches the layout of the license plate in Figure 5.

*Figure 6 (Gupta, Purohit, & Rathore, 2014)*

Cette méthode est décrite par Pratishtha Gupta, G N Purohit et Manisha Rathore, dans leur publication "Number Plate Extraction using Template Matching Technique" (Gupta, Purohit, & Rathore, 2014).

## b. Optical Character Recognition (OCR)

La fonction Matlab ocr permet de reconnaître du texte par la reconnaissance optique de caractères. Elle fonctionne en plusieurs étapes :

- les régions contenant du texte sont localisées, après que chaque symbole ait été extrait suite à une segmentation.
- les symboles extraits sont traités pour éliminer le maximum de bruit possible.
- on identifie chaque symbole grâce à la comparaison de ces symboles extraits avec des templates collectés.
- les symboles reconnus sont utilisés pour former les mots et nombres présents sur l'image de départ.



Figure 7 (Kumar Singh, Gupta, & Saxena, 2016)

Cette méthode est décrite par Ankit Kumar Singh, Aman Gupta, Aman Saxena, dans leur publication "Optical Character Recognition: A Review" (Kumar Singh, Gupta, & Saxena, 2016).

## II. Description détaillée de la méthode utilisée

### 1) Détection de la plaque

La première étape de la détection consiste à ouvrir l'image de la plaque au format RGB, c'est-à-dire sous forme de matrice, qui pour chaque pixel prends 3 valeurs : la valeur de la couleur rouge, la valeur de la couleur verte et la valeur bleue. Or, les plaques françaises ont la particularité de posséder deux zones bleues rectangulaires entourant deux zones blanches. L'idée est donc de faire ressortir les zones bleues en soustrayant la matrice de la dimension bleue par la matrice image :

$$B - RGB$$



Les résultats obtenus sont très concluants pour la majorité des voitures où les zones bleues apparaissent alors orange, alors que le reste de l'image est très sombre. En revanche, pour les voitures blanches ou bleues, la différence entre le fond et les zones bleues est beaucoup moins contrastée.

Pour faciliter la suite du traitement de l'image, il faut passer l'image en nuances de gris pour que chaque pixel ait une seule valeur au lieu de trois. Pour cela, on applique l'équation suivante :

$$Y = 0,229R + 0,587G + 0,114B$$

Une fois l'image passée en niveaux de gris, on applique un filtre de Canny. Le filtre de Canny est basé sur trois différentes étapes. Tout d'abord, on réduit le bruit grâce à un filtre gaussien 2D qui a pour formule :

$$g(x, y) = \frac{1}{\sqrt{2\pi} \sigma} \times e^{-\frac{x^2+y^2}{2\sigma^2}}$$

Où  $\sigma$ , l'écart type, joue le rôle d'échelle qui définit ce qui peut être gardé ou pas, par rapport à la taille des objets. Plus  $\sigma$  est grand, plus le lissage est important.



Ensuite, on cherche à calculer le gradient qui permet d'avoir l'intensité des contours. Pour cela, on applique 2 masques à l'image :

$$G_x = \begin{bmatrix} -1 & 0 & 1 \end{bmatrix} ; G_y = \begin{bmatrix} -1 \\ 0 \\ 1 \end{bmatrix}$$

Les non-maxima locaux sont supprimés. Enfin, on applique un seuillage grâce à deux seuils. Si les valeurs sont au-dessus du seuil le plus haut, elles sont conservées ; si elles sont entre les deux seuils, elles sont conservées si elles sont voisines de valeurs déjà conservées, sinon elles sont supprimées.



On réalise une fermeture par un élément structurant de taille 40. Une fermeture est une dilatation suivie d'une érosion par un élément B. Elle consiste à combler les trous et mieux relier les objets entre eux.

Une dilatation consiste en une extensivité par l'élément structurant si le centre de l'élément est compris dans l'objet. Elle se résume par l'équation suivante :

$$D(X, B) = X \oplus \hat{B} = \{y | y = x + b; x \in X, b \in \hat{B}\}$$

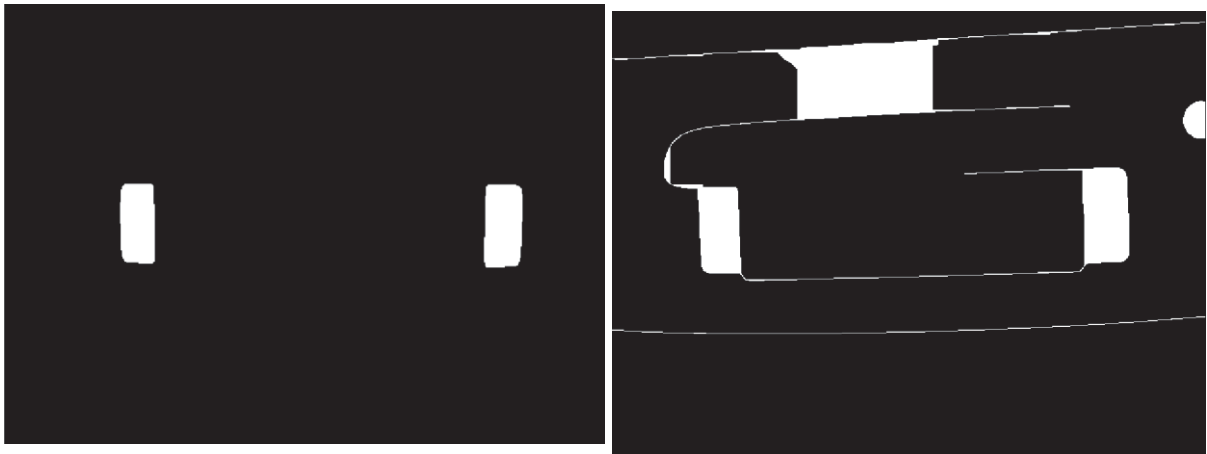
Une érosion consiste en une réduction par l'élément structurant, à l'exception du centre, si l'élément est compris dans l'objet. Elle se résume par l'équation suivante :

$$E(X, B) = \{y | B_y \subset X\}$$

L'équation de la fermeture s'écrit donc :

$$X \bullet B = E(D(X, B), \hat{B})$$

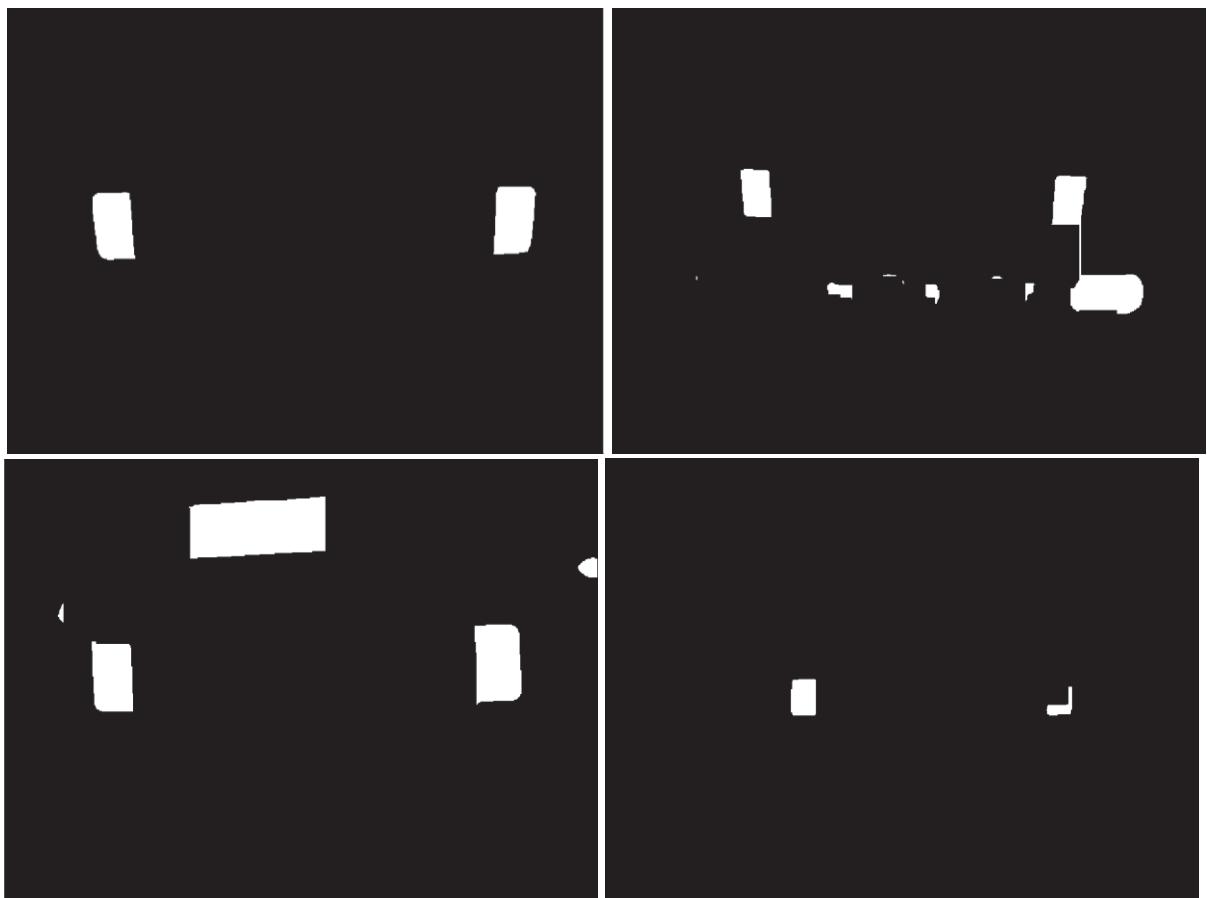
On obtient donc les résultats suivants :



Pour certaines images où le bleu était bien contrasté, deux carrés blancs ressortent bien. Néanmoins, certaines images qui étaient moins contrastées contiennent beaucoup de lignes blanches horizontales qui ne font pas partie de la plaque.

Pour les éliminer, on va donc appliquer une érosion sur un élément structurant qui correspond à une ligne horizontale.

On obtient donc les résultats suivants :



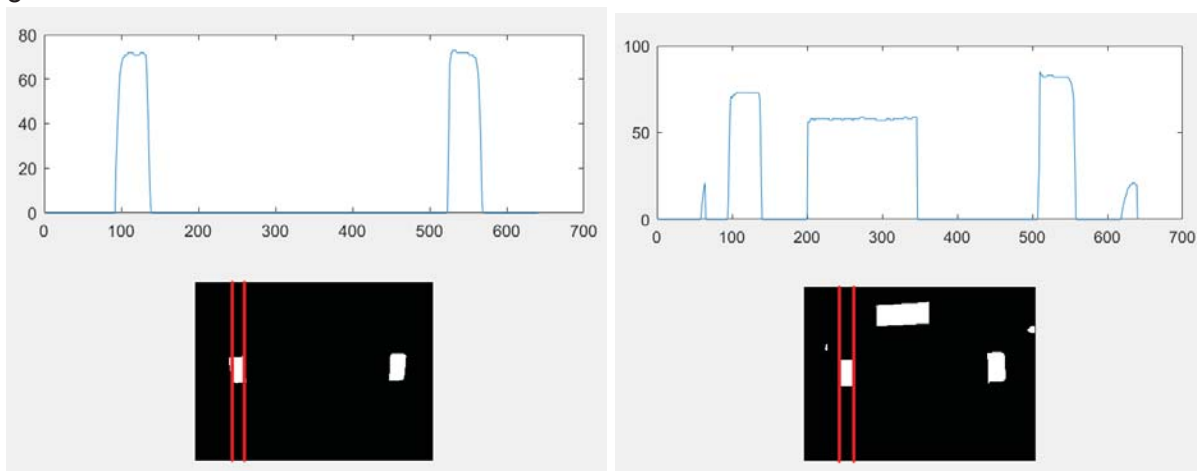
La plupart des résultats contiennent deux carrés blancs de la taille des carrés bleus de la plaque. Néanmoins, sur certaines images, les carrés ont été en partie effacés, ou alors certains éléments ne faisant pas partie de la plaque ressortent quand même.

L'étape suivante consiste à calculer l'histogramme cumulé horizontal sur la moitié de l'image pour trouver le carré blanc de gauche. En effet, la partie droite de l'image contient, pour beaucoup de photo, du bruit.

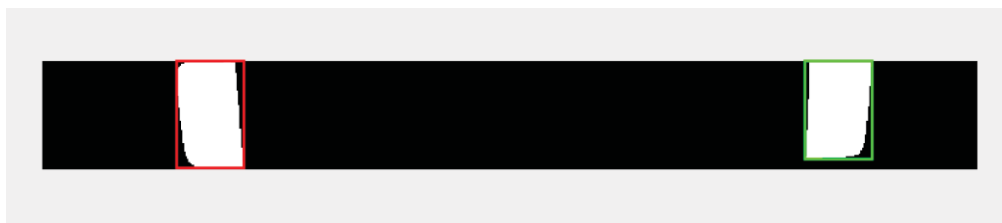
Un histogramme cumulé vertical consiste à faire la somme de la valeur des pixels sur chaque colonne :

$$p(x_0) = \sum_{y=0}^{W-1} f(x_0, y), 0 \leq x_0 < H$$

Un seuil à 9/10 de la valeur maximale permet de déterminer la largeur du carré blanc de gauche :



On recadre l'image sur la largeur trouvée, et on cherche les propriétés des composantes connexes, notamment le rectangle encadrant tous les composants connexes. Cela permet de déterminer la hauteur à laquelle se trouve le rectangle. Grâce à cette valeur, on peut recadrer l'image sur la hauteur pour trouver le deuxième rectangle. Enfin il reste à trouver les rectangles encadrants des composantes connexes pour déterminer la position exacte des deux carrés :



Comme on l'a vu précédemment, certains rectangles sont rognés sur la hauteur à cause des opérations de filtrage et d'érosion. Néanmoins, on peut supposer le reste des contours grâce au fait que toutes les plaques d'immatriculations françaises doivent avoir la même taille qui est de 52x11cm.

Il ne reste qu'à encadrer la plaque grâce aux différents paramètres calculés précédemment :



## 2) Détection des caractères

Une fois les plaques détournées, il faut déterminer précisément la position du texte afin de l'analyser et le reconnaître.

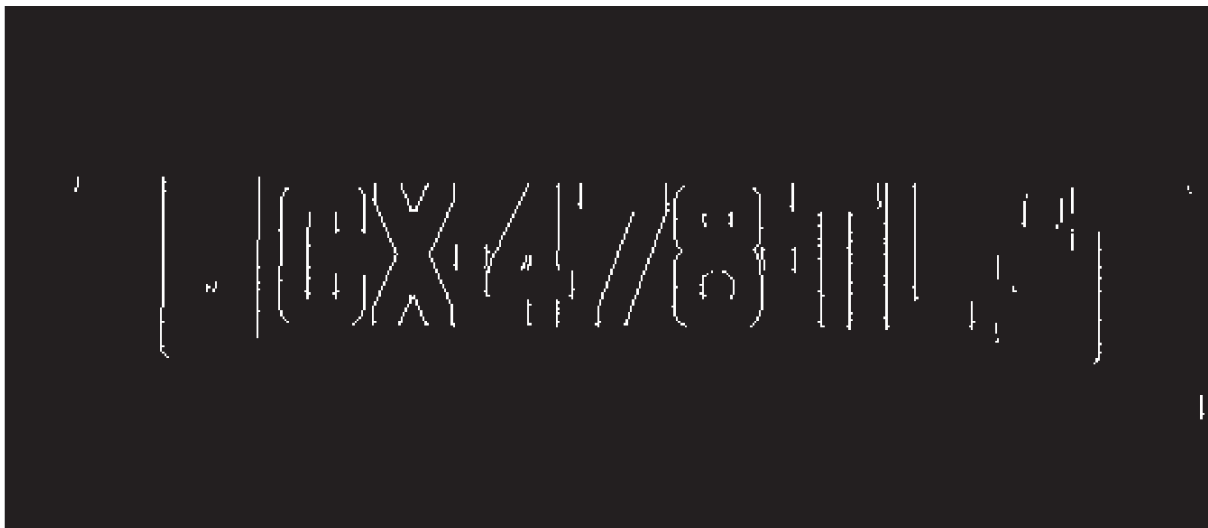
On commence par recadrer l'image autour de la plaque, puis on applique un filtre médian qui permet d'enlever le bruit, en remplaçant chaque pixel par la valeur médiane de son voisinage. On réalise ensuite une fermeture par un élément structurant de  $x$  pixels afin d'enlever les petits éléments noirs pour ne pas créer de faux positifs.



On applique une érosion pour renforcer les structures noires, ici les chiffres et les lettres de la plaque.



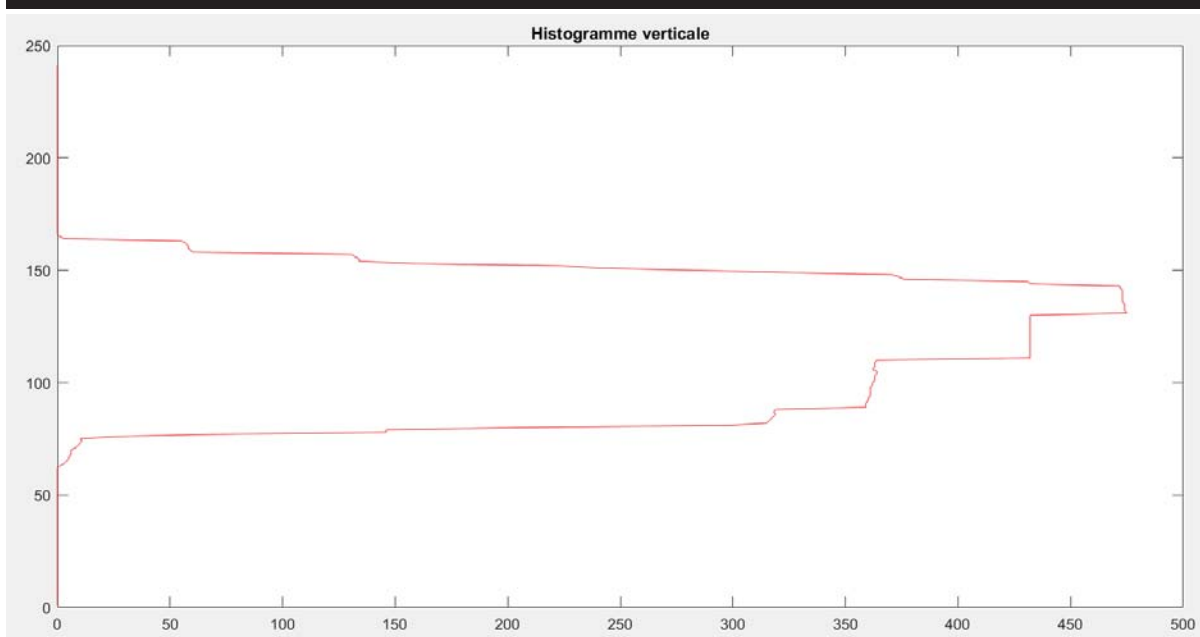
On passe ensuite par un filtre de Canny pour ne conserver que les contours verticaux :



Enfin, on réalise une fermeture suivie d'un histogramme cumulé horizontal afin de connaître la hauteur des lettres.

L'historgramme cumulé vertical est la somme de la valeur des pixels pour chaque ligne. On la modélise par l'équation suivante :

$$p(y_0) = \sum_{x=0}^{H-1} f(x, y_0), 0 \leq y_0 < W$$



Le texte peut donc être encadré grâce à un seuil égal à  $H/2$ .



Une fois le texte recadré, il faut faire ressortir les lettres par rapport au fond de l'image, pour que quand l'image sera binarisée, seuls les caractères seront blancs. On applique donc un Bottom Hat, qui permet de garder les valeurs sombres de l'image.

Un Bottom Hat consiste à soustraire l'image fermée par l'image originale :

$$(f \bullet B) - f$$



On peut alors passer l'image en binaire. Grâce à un seuil fixé, les pixels prennent soit la valeur 1 soit la valeur 0 en fonction de leur valeur initiale. On applique enfin une ouverture afin de supprimer les petits objets blancs qui ne seraient pas des caractères de la plaque. Une ouverture consiste en une érosion suivie d'une dilatation. Elle est donc l'inverse de la fermeture. L'équation qui la modélise est la suivante :

$$X \circ B = D(E(X, B), \hat{B})$$



### 3) Reconnaissance des caractères

Maintenant que les caractères sont correctement encadrés et binarisés, on peut faire la corrélation avec une base modèle.

0 1 2 3 4 5 6 7 8 9



Ces chiffres ont été préalablement recadrés afin de créer 10 images de même taille de chacun des chiffres :



Une fois ouverts dans Matlab, ils sont binarisés et inversés pour que les chiffres soient en blanc et le fond en noir. On calcule ensuite un coefficient pour les mettre à la même échelle que les chiffres de la plaque :

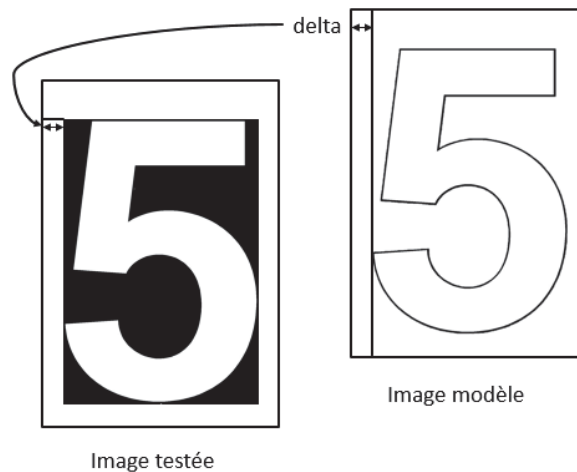
$$scale = HauteurPlaque / HauteurChiffres$$

On applique ensuite ce coefficient à l'image du chiffre pour changer sa taille de la manière suivante :

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} scale & 0 & 0 \\ 0 & scale & 0 \\ 0 & 0 & 1 \end{bmatrix} \times \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

On labélise ensuite chaque composant connexe de l'image de la plaque, c'est-à-dire que l'on met une étiquette différente pour les pixels n'appartenant pas au même groupe, avec pour condition d'appartenance, le voisinage à 8.

Pour chaque objet labélisé, on crée une nouvelle image qui a la même hauteur que l'image originale, et dont la largeur commence aux coordonnées de l'objet, moins un delta, et qui va jusqu'à sa largeur plus un delta. Delta correspond à la bordure présente dans l'image du chiffre modèle, entre le bord et le caractère.



On calcule ensuite la corrélation entre cette nouvelle image et chaque image des chiffres modèles.

Pour cela on utilise la formule de la corrélation normalisée :

$$c(s, t) = \frac{\sum_x \sum_y [f(x + s, y + t) - \bar{f}(s, t)] [w(x, y) - \bar{w}]}{\sqrt{\sum_x \sum_y [f(x + s, y + t) - \bar{f}(s, t)]^2 \sum_x \sum_y [w(x, y) - \bar{w}]^2}}$$



Avec la moyenne des pixels de l'image à analyser, la moyenne de l'image modèle et les coefficients de correspondance compris entre 0 et 1. Pour chaque chiffre de la plaque, on déduit à partir du coefficient le plus élevé quelle image référence correspond le mieux, et donc de quel chiffre il s'agit.

On applique une deuxième méthode pour reconnaître les chiffres : il s'agit de la méthode OCR. Cette méthode s'applique en trois étapes. Tout d'abord, elle fait un prétraitement sur l'image pour enlever le bruit et redresser le texte s'il n'est pas droit. Puis, elle segmente les lignes et les caractères. Enfin, elle reconnaît les caractères soit en les comparant à une base préenregistrée, soit en étudiant leurs caractéristiques.

### III. Difficultés rencontrées et évaluation des résultats

Les deux principales étapes de ce projet étaient de détourner les différents composants de la plaque et de reconnaître les chiffres au centre de la plaque.

La plupart des difficultés venait du fait que certaines voitures sur les photos possèdent des motifs, en particulier un des véhicules possède un autocollant de numéros de téléphone, ce qui peut créer beaucoup de faux positifs.

De plus, la couleur des voitures a également été problématique, entre les voitures blanches qui ont peu de contraste avec la plaque d'immatriculation, et les voitures bleues dont la couleur se confond avec les deux rectangles bleus de la plaque d'immatriculation.

Pour la première étape, la position de la plaque a été correctement détectée pour l'ensemble des photos. Il existe quelques détections qui pourraient être améliorées, notamment pour les plaques qui ne sont pas parfaitement droites.

Pour la deuxième étape, on obtient 72,12% de réponses correctes avec la reconnaissance par corrélation, et 75,15% de réponses correctes avec la reconnaissance par OCR, ce qui n'est pas suffisant. Pour améliorer ces résultats, il faudrait pouvoir faire la corrélation avec plus de polices de caractères différents, mais également corriger le fait que certains caractères soient coupés ou légèrement penchés. On aurait également pu mettre en place un réseau de neurones qui aurait eu une analyse beaucoup plus fine et aurait aussi pu prendre en compte la segmentation de la plaque dans l'image.



En revanche, pour la détection des lettres et des chiffres, on n'obtient que 35% de réussite, contre 72% avec la méthode OCR. Cela est en partie dû à un mauvais encadrement des caractères qui servent à la corrélation.

# Conclusion

Le but de ce projet était de détourer les plaques d'immatriculation et reconnaître les chiffres et les lettres de la plaque à partir de photos de voitures. Il a été l'occasion de mettre en pratique différentes techniques de traitement d'images, abordées tout au long de l'année en vision par ordinateur. Il utilise notamment les techniques de morphologie mathématiques, de filtrage, de transformations géométriques, de segmentation et de reconnaissance de formes. Il a également été l'occasion de parcourir la littérature scientifique sur ces différents sujets.

Malgré certains résultats approximatifs, notamment dans le traitement des lettres, le programme est fonctionnel, et permet par exemple de détourer la plaque précisément pour toutes les photos.

Des améliorations sont possibles pour un fonctionnement casi parfait, et cela pourrait être un sujet de recherche plus poussé, abordant par exemple les techniques de Deep-Learning, aujourd'hui beaucoup plus fiable pour les image 2D, que les méthodes de traitement d'image traditionnelles.

# Bibliographie

- CHHAYDER, A., & BELHADJ MOHAMED, I. (2009, Mars 26). *Système de Reconnaissance Automatique des Plaques*. Récupéré sur 5th International Conference: Sciences of Electronic, Technologies of Information and Telecommunications :  
[http://www.setit.rnu.tn/last\\_edition/setit2009/Image%20and%20Video/18.pdf](http://www.setit.rnu.tn/last_edition/setit2009/Image%20and%20Video/18.pdf)
- CLOUARD, R. (2007, mai 24). *TUTORIEL : DÉTECTER DES PLAQUES D'IMMATRICULATION*. Récupéré sur Le Projet Panthéon:  
<https://clouard.users.greyc.fr/Pantheon/experiments/licenseplate-detection/index-fr.html>
- Gupta, P., Purohit, G. N., & Rathore, M. (2014, Février). *Number Plate Extraction using Template Matching*. Récupéré sur Semantic Scholar.
- Kumar Singh, A., Gupta, A., & Saxena, A. (2016, Avril). *Optical Character Recognition: A Review*. Récupéré sur Semantic Scholar:  
<https://pdfs.semanticscholar.org/0fd3/a2b4c2cfc2c812243a8661993a21f17b3411.pdf>
- KUMAR, E. M. (2018, Avril 8). *Project Report on Automatic Number Plate Recognition using Matlab*. Récupéré sur colorsglobal: <https://www.colorsglobal.com/2017/03/matlab-image-processing-car-number-plate-detector-project-code-report.html>
- Towards DataScience. (2017, Aout 11). *Number plate detection with Supervisely and Tensorflow (Part 1)*. Récupéré sur Towards DataScience:  
<https://towardsdatascience.com/number-plate-detection-with-supervisely-and-tensorflow-part-1-e84c74d4382c>