# …

# ….

# Modèle proposé

## Introduction

Après avoir localisé les plaques d’immatriculation dans une images de surculation routière, notre contribution consiste en trois étapes essentielles : i) préparation(prétraitement) des plaques localisées afin d’améliorer la qualité de l’image, ii) segmentation des caractères présents dans les plaques, iii) reconnaissance des caractères segmentés pour identifier les plaques.

La segmentation représente le défi(chalenge) principal dans notre projet où la technique de projection verticale et la technique de segmentation en composantes connexes sont les deux méthodes les plus utilisées dans les systèmes LPR. Notant aussi que les plaques d’immatriculation sont relativement normalisées (respectent certaines normes), par exemple, elles contiennent des segments de caractères alignés, appartenant à un jeu de caractères bien défini et occupent une position centrale dans la plaque. Ces informations a priori sur la structure des plaques dans les véhicules peuvent être exploitées pour améliorer la qualité de segmentation.

Bien évidement chaque technique a ses avantages et ces inconvénients, c’est-à-dire que l’une est plus efficace que les autres avec certaines plaques et moins efficace avec d’autres. Pour cette raison, nous envisageons de proposer un modèle de segmentation en essayant de bénéficier d’avantages des trois techniques précédentes, à savoir : projection verticale, analyse en composantes connexes et segmentation à base d’information a priori.

La projection verticale est simple, très rapide mais elle n’arrive pas à séparer les caractères inclinés et ceux qui sont fortement chevauchés. L’analyse en composante connexe est une technique très utilisée, elle n’est pas sensible ni à la rotation ni à la position des objets dans la plaque, mais elle très limitée dans la présence du chevauchement. La segmentation à base des information apriori(la taille de la plaque d’immatriculation, la taille d'un caractère, la taille de l'intervalle entre les caractères et le nombre de caractères dans la plaque) sur la structure des plaques est est beaucoup plus efficace que les deux premières méthodes dans le cas où la plaque est de mauvaise qualité(bruitée), néanmoins cette méthode est complétement erronée dans la présence des structures particulières tel que les plaque qui contiennent des signes publicitaires, séparateurs ou des espacements entre les caractères.

Il est nécessaire de concevoir un schéma optimisé et efficace pour la combinaison de tous ces techniques, dont l’objectif est d’améliorer les performances de la segmentation, et par conséquence ceux du système, sans engendrer une dégradation remarquable sur le temps d’exécution.

Les RBs sont des modelés graphiques et probabilistes qui sont mieux adaptés pour le problème de classification avec la présence des données incomplètes, voir chapitre II $ I.4. Ce chapitre est consacré à présenter notre solution qui sera une amélioration(continuation) des travaux antérieurs réalisé au sein de l’EMP.

## Travail antérieur à l’EMP

Dans le cadre des projets de fin d’étude à l’EMP, notre thème a été traité trois fois, dont la dernière où les résultats ont été relativement bons était dans le travail de l’année 2014-2015. La solution proposée dans ce travail [1] est développée avec le langage Matlab pour réaliser un module de segmentation des plaques d’immatriculation en Algérie. Elle est basé sur la méthode d’analyse de composantes connexes CCA pour segmenter la plaques et le classificateur CBN afin de minimiser les erreurs (les segment qui ne contiennent pas un caractère).

### Diagramme de la solution

L’algorithme proposé pour la segmentation des plaques d’immatriculation dans le projet de fin d’étude 2014-2015 est illustré dans le diagramme de la Figure ‎IVV.2.

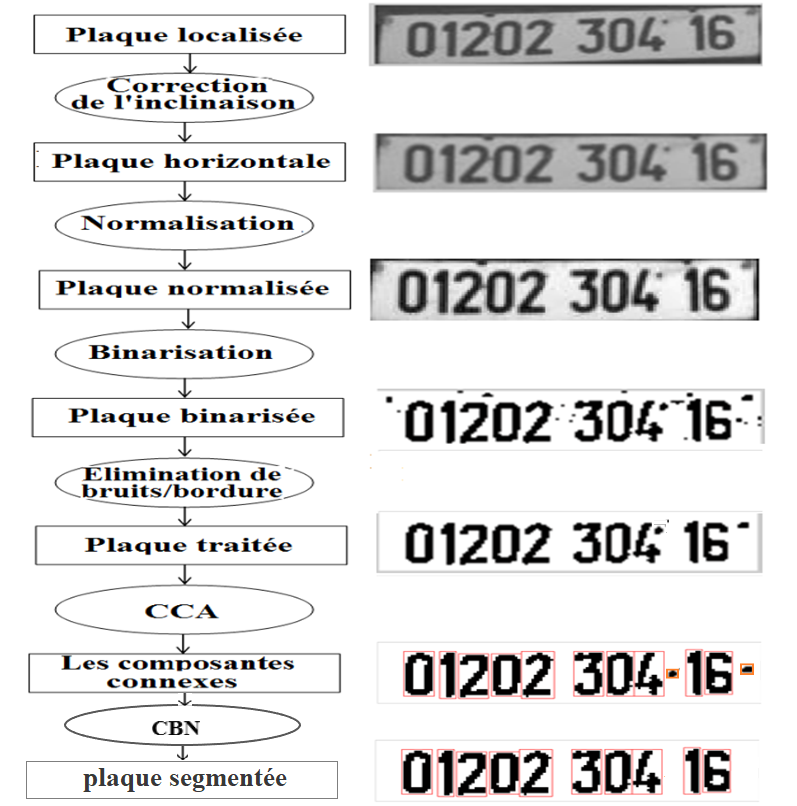


Figure ‎III.1 Diagramme de la solution proposée dans le PFE 2014-2015 à l’EMP.

Afin de détailler un peu le diagramme de solution ci-dessus nous allons expliquer brièvement les principales étapes :

1. Binarisation : ils ont utilisé le seuillage local de Sauvola.
2. CCA : Extraction des composantes connexes par l’algorithme de composantes connexes.
3. CBN : calculer certains attributs pour chaque composante connexe (tels que la hauteur, la largeur, la densité en pixels noirs… etc.) et l’appariement des attributs calculés avec les informations a priori afin de pouvoir distinguer entre deux classes(décisions) : les composantes de type bruit, et les composantes de type caractère.

### Avantages

* Insensible à l’inclinaison et à la résolution de la plaque.
* Possibilité d’isoler les objets seulement et non pas les régions contenant les objets, ce qui est efficace pour éviter l’extraction des parties d’objets voisins.
* Donne de bons résultats avec les plaques de bonne qualité.

### Limites

* Les caractères chevauchés qui sont souvent présents sont considérés comme un seul objet.
* Un caractère endommagé est considéré comme deux ou plusieurs objets.
* Ne donne pas des résultats satisfaisants avec les plaques de mauvaise qualité.

## Notre solution

### Améliorations

Afin de dépasser les limites citées ci-dessus ($ III.2.3) et améliorer les performances du système nous avons modifié les parties suivantes : en premier lieu, la méthode de prétraitement utilisée (seuillage locale de Sauvola) en utilisant le filtre unsharp suivi par le seuillage de Otsu qui ont donnée de meilleurs performances (…ici cité une figure …). En deuxième lieu, nous avons conçu une nouvelle méthode de segmentation en combinant trois techniques à savoir : la technique de segmentation en composantes connexes, la projection verticale et l’exploitation des informations à priori, en se basent sur le classifier BN. Après avoir effectué une segmentation en utilisant l’une des techniques, par exemple la projection verticale, pour chaque segment on choisit un ensemble de descripteurs(factures) où le calcule est rapide. Ces descripteurs vont être exploiter par le CBN pour prendre l’une des décisions suivantes : i) le segment représente un caractère, ii) le segment représente un bruit, iii) le segment contient plus d’un objet (par exemple un chevauchement de deux caractères), vi) aucune décision, du moins pour l’instant.

La combinaison est faite sous forme de chaine de responsabilité [1], c’est-à-dire, pour une méthode de segmentation donnée la prise de la troisième décision pour un segment quelconque veut dire que cette méthode n’a pas pu séparer entre les objets présents dans ce segment, par conséquent il est à la responsabilité de la méthode suivante, dont l’objectif final est de séparer est isoler tous et seulement les segments qui représentent un caractère et qui serons passés à l’étape de reconnaissance.

La dernière partie dans notre solution est l’étape de reconnaissance qui consiste à identifier le caractère présent dans chaque segment afin de reconnaitre la plaque entière. A ce niveau la précision du classifieur est prioritaire ainsi que sa rapidité, pour cette raison nous avons choisi de travailler avec la méthode SVM.

### Diagramme de la solution

Le figure III.2 montre le diagramme de notre solution qui peut être considérée comme étant une amélioration de la solution proposée dans le PFE 2014-2015 au sein de l’EMP voir la figure III.1.

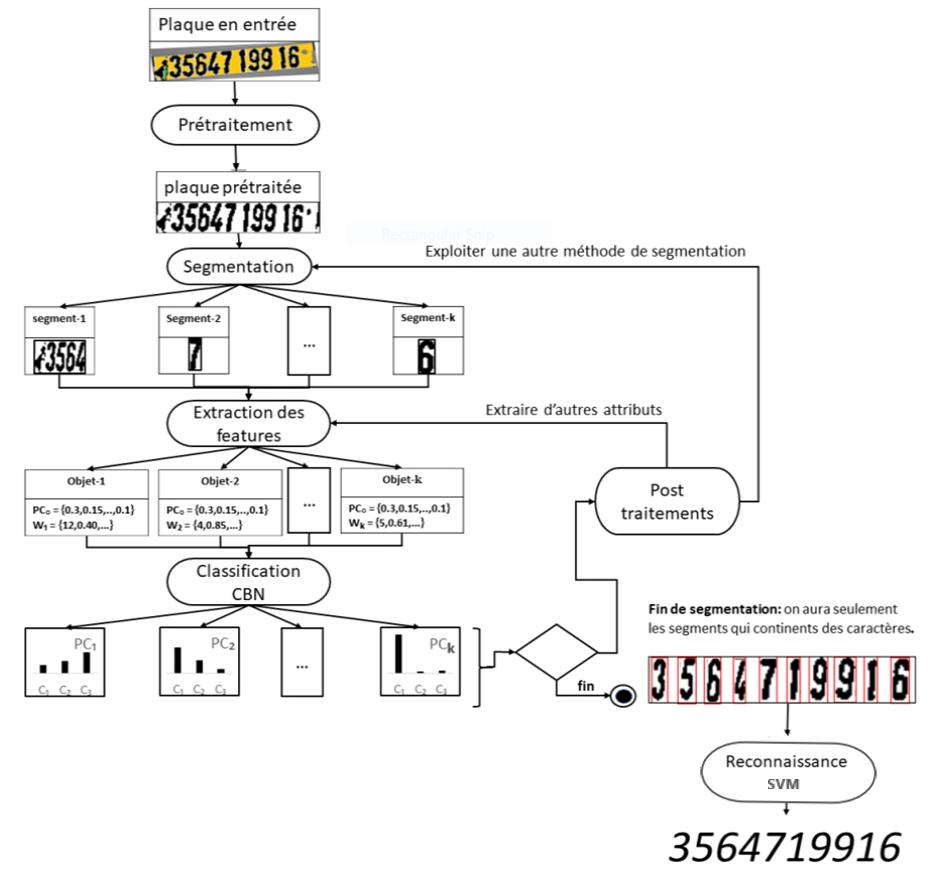


Figure ‎III.2 diagramme de la solution.

Dans les sections suivantes nous expliquerons en détailles les composantes notre solution que sont montrer dans le diagramme ci-dessus, voir la figure III.2.

### Prétraitement

Les plaques d’immatriculation dans une scène de circulation routière peuvent être inclinées, partiellement cachées, proches ou lointaines, éclairées ou ombrées, tachées et bien d’autres. La normalisation de tous ces facteurs est nécessaire pour faciliter le processus de segmentation d’une part et pour améliorer son ces résultats d’une autre part.

Nous informons que dans notre projet et dans toutes les sections qui suivent, nous travaillerons avec les couleurs niveau de gris (256 couleur), par conséquent toute image capturée sera transformée vers une image de ce type.

#### Correction de l’inclinaison

La plaque obtenue à la sortie de la localisation est très souvent inclinée par rapport à l’horizontale, cette inclinaison est dû aux diverses raisons à savoir : i) la position de la caméra par rapport au véhicule au moment de la capture, ii) erreurs dans le processus de la localisation.

La segmentation doit être non sensible à l’inclinaison ou à l‘échelle de la plaque localisée. Par conséquent, une normalisation géométrique est nécessaire pour avoir une plaque horizontale, de dimensions normalisées et similaire. Une solution efficace à ce problème consiste à appliquer une transformation géométrique appelée « transformation affine », voir la $ I.\*\*\*. Dans notre cas cette transformation sera composée de deux transformations élémentaires : une rotation pour rendre la plaque horizontale, suivi par une mise à l’échelle afin de normaliser les dimensions (largeur et hauteur), voir la figure III.3.

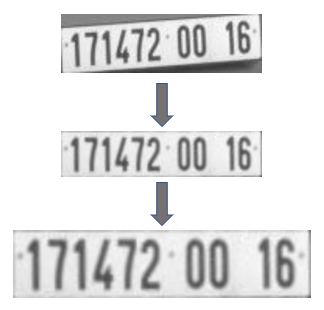


Figure III.3 Exemples de correction d’inclinaison avec la transformation affine.

#### Binarisation

La binarisation est un problème délicat dans le domaine de traitement d’images. Plusieurs thèses et recherches ont porté sur ce thème vu les différents domaines qui l’utilisent. Dans notre travail, après avoir implémenté et essayé plusieurs approches de seuillages (binarisation), parmi lesquelles sont évoqués dans la $ I.\*\*\*, nous avons constaté une erreur (limite) commune entre toutes ces approches et surtout celles du seuillage local. Cette erreur est constatée très souvent dans les régions entre caractères, c’est-à-dire des régions de pixels de l’arrière-plan résidant entre les caractères sont devenus des connexions (chevauchement) entre ces caractères après seuillage, voir la figure III.4.

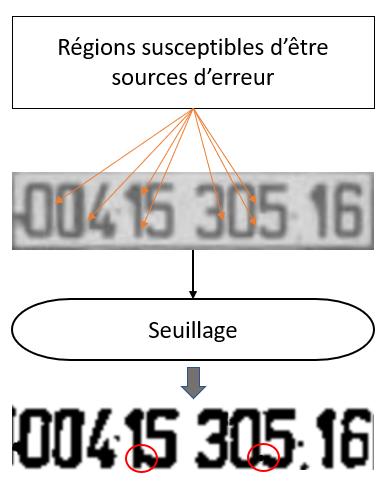


Figure III.4 problème de chevauchent des caractères après binarisation.

Ce problème comme il le montre la ci-dessus (figure III.4) n’est pas issu de la méthode de binarisation(seuillage) utilisée, mais plutôt à l’image elle-même, il n’y a pas de grande marge entre le niveau de gris des caractères et les régions indiquées comme étant des sources d’erreur, c’est-à-dire, il n’y a pas de frontières claires entre les caractères. Nous avons trouvé une bonne solution à ce problème qui consiste à étaler cette marge en appliquant un filtre d’aiguisage pour bien tracer les frontières de chaque caractère, pour cela nous avons utilisé le filtre unsharp (Unsharp filter), voir la $ I.\*\*\*. Un exemple démonstratif est illustré dans la figures suivante (figure III.5).

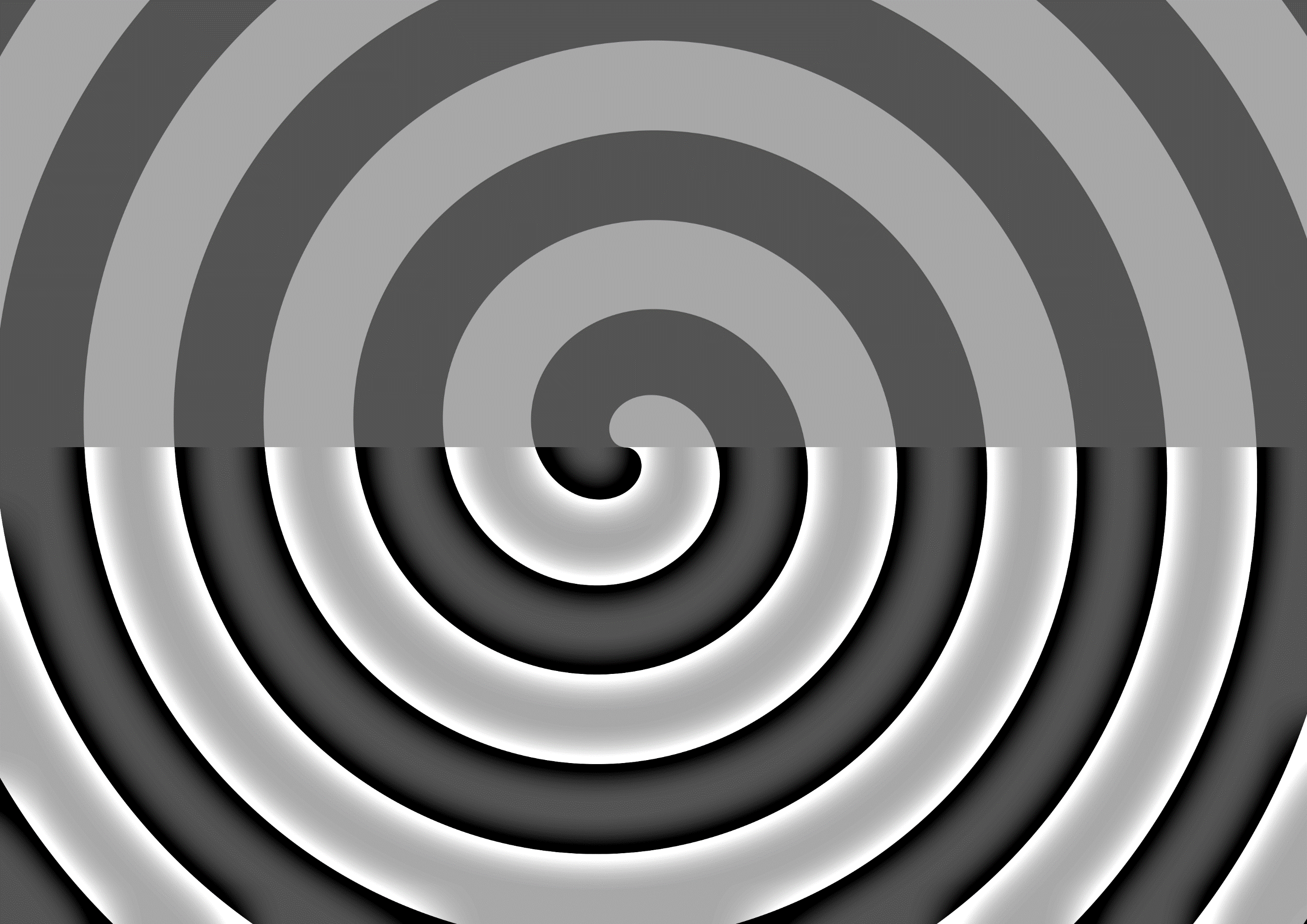


Figure III.5 exemple démonstratif du filtre unsharp (le filtre est appliqué sur le moitié inferieur).

La figure III.6 montre un exemple de binarisation après avoir applique le filtre unsharp, on remarque bien que ce filtre a pu corriger les erreurs présentées dans la figure III.4 (les chevauchements encerclés en rouge).

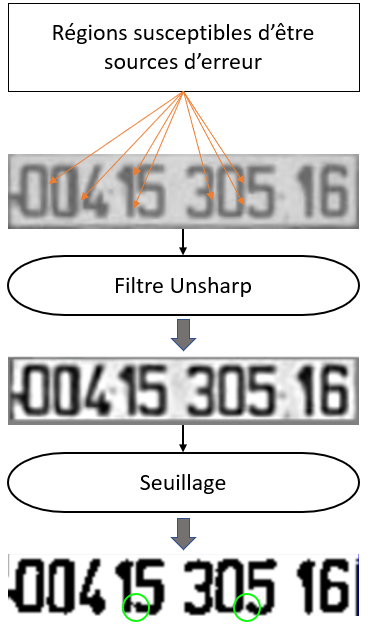


Figure III.6 exemple d’application du filtre unsharp.

Les essais appliqués aux différentes méthodes de seuillage combinés avec le filtre unsharp nous ont montré que le seuil de Otsu est le plus performent en termes de : préservation de l’information, minimisation du bruit et rapidité de calcule. Le seuil de Otsu est un seuil global optimal pour une image bimodale (une image où on trouve deux couleurs dominantes) comme le cas dans les plaques d’immatriculation. Il est à noter que le filtre unsharp est nécessaire pour que le seuillage de Otsu donne de bons résultats.

### Segmentation

Dans la section précédente nous avons affirmé que l’étape de prétraitement donne de très bons résultats, surtout quand il s’agit d’une plaque de bonne qualité, c’est-à-dire, un être humain peut facilement reconnaitre le matricule écrit dedans. Néanmoins, les données montre qu’on pourra toujours avoirs des plaques de mauvaise qualité (plaque tachée, sale, rayée ou de mouvais éclairage) où l’étape de prétraitement (binarisation) peut fournir des résultat erronés (présence du bruits, caractères chevauché ou endommagés). Les essais appliqués sur un échantillon de plaques réelles ont montré que l’occurrence de ces cas de figures est considérable. Pour cette raison dans notre projet nous avons pris en compte la possibilité d’avoir des plaques de qualité dégradée. Afin d’améliorer les performances du processus de segmentation et par conséquent de notre système nous avons conçu la solution schématisée dans la figure II.2 Pour bien l’expliquer, une plaque (résultat du prétraitement ) en entrée sera traitée comme suite :

1. Segmentation de la plaque par la méthode des projections en plusieurs segments.
2. Extractions de certains attributs pour chaque segment (tels que la hauteur, la largeur, la densité en pixels noirs, et d’autres) ces attributs doivent être rapides à extraire.
3. Appariement des attributs mesurés avec les informations (probabilités) a priori en utilisant un classifier BN afin de pouvoir distinguer (décider) entre les segments qui contiennent du bruit, caractère, ou un chevauchement de caractères (un segment relativement large). A ce niveau-là, la décision suit un raisonnement probabiliste, mais avec contrainte pour être plus sûr, c’est-à-dire pour attribuer un segment à une des trois classe, la probabilité pour cette classe doit être la plus grande et dépasse un certain seuil(contrainte), par défaut le seuil est 75%.
4. Les segments(objets) vont être filtrés comme suite :
5. Un objet classé comme bruit sera ignoré(supprimer).
6. Un objet classé comme caractère va être marqué comme caractère.
7. Un objet classé comme étant un chevauchement sera marqué comme chevauchement et passé à l’étape post traitements.
8. Un objet indécis (non décidé) sera passé à l’étape post traitements.

Si à un moment donné aucun des segments n’est marqué comme indécis ou comme étant un chevauchement, le processus de segmentation s’arrête pour cette plaque.

1. Post traitements, cette étape reçois en entrée deux type de segments (objets) : un segment indécis ou un segment qui présente un chevauchement.

a) Attributs faciles : calculés aisément comme la longueur, la largeur et la densité en pixels. b) Attributs couteux : demandent beaucoup de calcul comme le score d’appariement entre une composante et les modèles de référence des caractères. c) Attributs dépendants : ont besoin des résultats de segmentation des autres composantes de la plaque.

Il est important de souligner le fait que la connaissance de la valeur d’un attribut peut seulement contribuer dans la modification (augmentation ou diminution) de la probabilité que la composante représente un caractère. En plus, seulement une partie des attributs est disponible à un moment donné.

Réellement, nous sommes en train de faire une classification avec incertitude sur des données manquantes. Les réseaux Bayésiens (RBs) (voir chapitre III) constituent une solution satisfaisante à ce problème. Puisque les attributs sont approximativement indépendants étant donné la classe, nous avons opté pour une classification Bayésienne naïve (CBN) pour pouvoir décider la classe d’une composante connexe à partir de ses attributs.

Le pseudo-algorithme de segmentation que nous allons proposer est basé sur la méthode CCA et le classificateur CBN. Cet algorithme représente une amélioration importante de la première solution en utilisant seulement CCA, évoquée dans la section IV.4.

La construction d’un classificateur CBN est réalisée comme suit :

1) Déterminer un ensemble 𝐴 d’attributs discriminants qui sont utiles pour décider si une composante représente un caractère ou un bruit. 2) Construire un CBN sur l’ensemble d’attributs proposés, cela revient à : a. Discrétiser les domaines des variables (attributs). b. Estimer les tables de probabilités conditionnelles 𝑃(𝐴𝑖|𝐶). 3) Déterminer empiriquement deux seuils de décision: s1 et s2. La fonction de décision est la suivante :

𝑐𝑙𝑎𝑠𝑠𝑒\_𝑐𝑖 = {

𝑐𝑎𝑟𝑎𝑐𝑡è𝑟𝑒 𝑠𝑖 𝑃(𝑐𝑖 = 𝑐𝑎𝑟|𝐴) > 𝑠2 𝑖𝑛𝑑é𝑐𝑖𝑠 𝑠𝑖 𝑠1 ≤ 𝑃(𝑐𝑖 = 𝑐𝑎𝑟|𝐴) ≤ 𝑏𝑟𝑢𝑖𝑡 𝑠𝑖 𝑃(𝑐𝑖 = 𝑐𝑎𝑟|𝐴) < 𝑠1

𝑠2 ( IV.3 )

La variable classe\_ci peut prendre trois valeurs : ‘caractère’, ‘bruit’, ‘indécis’. La classe ‘indécis’ est prise lorsque les attributs retenus ne peuvent pas trancher si 𝑐𝑖 représente un caractère ou un bruit.

#### Classification Bayésien naïve

#### Post traitement

### Reconnaissance

Input+ methose svm +output

## Conclusion

…

[1]. Badenhorst W. (2017) Chain of Responsibility Pattern. In: Practical Python Design Patterns. Apress, Berkeley, CA