

Rapport de Mi-Parcours  
InSite  
Janvier 2018

Martin OLIVIER, Fabien GOGLIO



Suiveur :  
Sylvie ZAGO

## Table des matières

<b>1</b>	<b>Remerciement</b>	<b>1</b>
<b>2</b>	<b>Introduction</b>	<b>2</b>
<b>3</b>	<b>Présentation</b>	<b>3</b>
3.1	Description . . . . .	3
3.2	Objectifs . . . . .	3
<b>4</b>	<b>Cadrage</b>	<b>4</b>
4.1	Budget . . . . .	4
4.2	Dates clefs . . . . .	4
4.3	Organisation . . . . .	4
4.3.1	Démarche . . . . .	4
4.4	Planning . . . . .	4
4.5	Langages et technologies utilise . . . . .	4
<b>5</b>	<b>Démarche Initiale</b>	<b>5</b>
5.1	Relevés géomagnétiques et origine de notre dataset . . . . .	5
5.2	Détections d'objets . . . . .	6
5.2.1	Filtrage . . . . .	6
5.2.2	Algorithmes de détections de bords . . . . .	6
5.2.3	Problèmes rencontres . . . . .	11
<b>6</b>	<b>Développement d'une nouvelle méthodologie</b>	<b>12</b>
6.1	Détections d'objet metallique par réseaux neuronaux . . . . .	12
6.1.1	Caractérisation . . . . .	12
6.1.2	Explication de l'approche . . . . .	13
6.1.3	Problèmes initiaux . . . . .	13
6.1.4	Solutions . . . . .	14
<b>7</b>	<b>Bilan provisoire et perspectives</b>	<b>16</b>
7.1	Conclusion . . . . .	16
7.2	Implémentation d'un réseau de convolution . . . . .	16
7.3	Choix de la direction de recherche . . . . .	16
<b>8</b>	<b>Annexes</b>	<b>18</b>

---

# 1 Remerciement

Nous tenons a remercier

...

## **2 Introduction**

## **3 Présentation**

### **3.1 Description**

InSite analyse et détecte des objets archéologiques dans des relevés géomagnétiques, en utilisant des techniques novatrices dans le domaine de la Computer Vision, ainsi que de l'apprentissage automatique.

### **3.2 Objectifs**

Nous cherchons à :

- Analyser automatiquement des relevés géomagnétiques
- Produire des cartes d'intérêts, en indiquant les zones où des objets archéologiques pourraient se trouver
- Détecter et réduire le bruit présent dans l'image

## 4 Cadrage

### 4.1 Budget

Étant d'abord et surtout un projet de recherche d'informatique, InSite ne requiert pas de budget. Nous avons pu obtenir les relevés magnéto-métrique sans cout, et les ressources nécessaires pour mener a bien ce projet sont déjà a notre disposition.

### 4.2 Dates clefs

### 4.3 Organisation

InSite se compose de 2 étudiants, Fabien GOGLIO et Martin OLIVIER. Par cette taille réduite, l'organisation est simplifiée. Les rôles sont distribués selon l'intérêt de pour la tâche, et selon sa compétence a mener cette tâche a bien. Les recherches sur les techniques a utiliser, et le développement d'outil et de code sont faites de manières indépendantes, même si une communication constante est faite.

#### 4.3.1 Démarche

Comme inSite est un projet non pas de développement, mais de recherche, nous utilisons une démarche différentes de celle trouve habituellement dans les PST : nous passons une grande partie de notre temps a faire des recherches, et nous n'écrivons que des petits scripts, qui répondent a une tâche particulière.

### 4.4 Planning

### 4.5 Langages et technologies utilise

Pour simplifier le développement, nous utilisons Python2 avec les librairies numpy, pyplot, cv2, qui possèdent de nombreuses fonctions mathématique et d'analyse d'image déjà implémentées.

---

## 5 Démarche Initiale

### 5.1 Relevés géomagnétiques et origine de notre dataset

Les relevés géomagnétiques sont utilisés en archéologie depuis une vingtaine d'années. Ils permettent d'obtenir une idée de la répartition d'objets archéologiques intéressants, et est surtout une approche non destructive : elle ne modifie pas les objets qu'elle observe. Un tel relevé s'obtient en mesurant le magnétisme du sol, grâce à un gradiomètre magnétique, qui mesure la différence entre deux mesures du champ magnétique faites par deux capteurs. On construit ensuite des cartes en noir et blanc, représentant les variations du champ magnétique par une variation de couleur. Une valeur positive forte sera blanche, tandis qu'une valeur négative forte sera noire. Ces relevés permettent de détecter des structures et objets enfouis jusqu'à quelques mètres de profondeur. En effet lorsque l'on construit, ou lorsque l'on fait cuire à l'aide de fours, on modifie les valeurs de champs magnétiques du terrain, et ces variations peuvent être observées. Ces relevés, mesurant le magnétisme du sol sont également très sensibles aux objets métalliques, et le sont d'autant plus que ces objets sont proches du capteur, ce qui peut poser problèmes dans les zones où une activité anthropologique récente a lieu, car l'homme moderne rejette beaucoup de déchets métalliques. Ces objets métalliques contemporains sont des dipôles et provoquent sur les relevés des anomalies très caractéristiques. Nous discuterons de ces anomalies plus bas.

Nos relevés nous proviennent tous du site archéologique de Marsal, en Moselle, et ont été réalisés au cours d'une quinzaine d'années. Cette région possède des marais salants, et on trouve des traces d'une exploitation industrielle très importante aux alentours de l'âge du Fer (VII<sup>ème</sup>-I<sup>ère</sup> siècle avant J.C). On trouve des traces de ces exploitations avec par exemple une quantité extrêmement élevée de briquetage (déchet de terre cuite, utilisé pour la cuisson de la saumure) ; ce n'est pas moins de 4 millions de mètres cubes de briquetage accumulés dans la région, soit presque deux fois le volume de la pyramide de Khéops. Le sel était, et a longtemps été une ressource stratégique ; car en plus de servir d'épice, le sel est utilisé dans la conservation d'aliments, et les populations agricultrices, qui consomment moins de viande doivent compléter leur apport par du sel. La région a donc été très riche par le passé. Dans les fouilles, on trouve nombre de fours de grande taille, en forme de "U", qui servaient à la préparation du sel, et les corps trouvés sur les sites de productions de sel sont mal nourris, ce qui semble indiquer qu'ils sont des esclaves, ou en tout cas des "employés" qui ne possèdent pas les fours ni le sel qu'ils produisent.

---

## 5.2 Détections d'objets

Nous avons décidé de débiter nos recherches de détections de formes par des solutions simples, avec l'attente qu'elle nous donnent des résultats de bonne qualité. L'attrait de ces méthodes étaient principalement qu'elles ne requiert pas de quantité de données importante, et qu'elles étaient à notre porte d'un point de vue académique. Nous avons donc approché le problème de deux façons différentes :

- Filtrage
- Algorithmes de détections de bords.

### 5.2.1 Filtrage

TODO : REMPLIR

### 5.2.2 Algorithmes de détections de bords

Nous avons débuté par l'utilisation de l'opérateur de Sobel[7]. Cette technique produit des images en noir et blanc où les bords ont une valeur de blanc élevé. Un bord est défini comme un endroit de l'image où la magnitude de son gradient est élevée. Cette technique est très couramment utilisée en *Computer Vision* et il est déjà implémenté dans la librairie openCV en Python. Son fonctionnement est décrit en plus de détails ci-dessous :

**Recherche du gradient d'intensité de l'image** On applique un kernel de Sobel : il s'agit simplement de deux matrices 3x3 permettant de calculer la dérivée première verticale et horizontale par convolution. Ces deux matrices sont :

$$G_x = \begin{bmatrix} +1 & 0 & -1 \\ +2 & 0 & -2 \\ +1 & 0 & -1 \end{bmatrix} \quad G_y = \begin{bmatrix} +1 & +2 & +1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix}$$

A chaque point de l'image on détermine la magnitude de ce gradient par la formule

$$\nabla(G) = \sqrt{G_x^2 + G_y^2}$$

**Création d'une image des bords** Une fois cette opération réalisée, il nous suffit d'associer la valeur du gradient à une valeur de gris. Une grande magnitude du gradient, indiquant que le pixel original appartenait à un bord, donnera un pixel blanc, tandis qu'une faible intensité donnera un pixel noir. Sur des images "propres" on obtient ce genre de résultats :

---



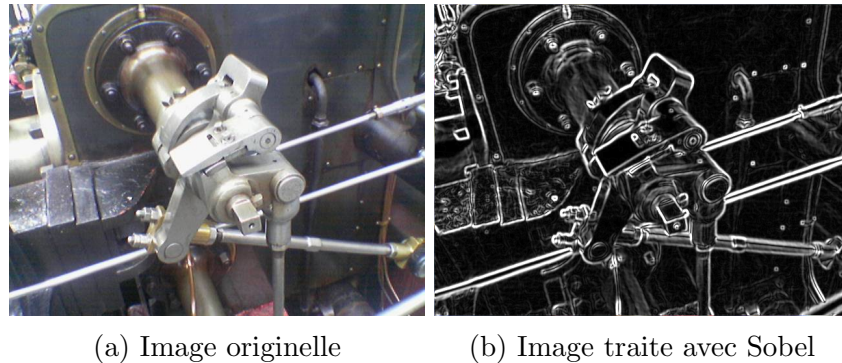


FIGURE 1 – Comparaison d'une image sans traitement avec celle traitée avec Sobel, de Wikipedia [5][6]

On observe que les détails de l'image disparaissent, et que les bords, c'est à dire les endroits où la magnitude du gradient de l'image est importante, sont visibles par des nuances de gris.

Lorsque l'on applique l'opérateur Sobel sur nos images, nous obtenons, dans les meilleurs des cas, ces résultats :

Sur nos images, les résultats obtenus ne sont pas très pertinents. Le bruit est trop prévalent, et les bords des objets que nous cherchons à détecter sont trop faibles pour que l'opérateur de Sobel puisse donner des résultats intéressants.

Nous avons ensuite décidé d'utiliser un autre algorithme, celui de Canny [1]. L'algorithme de Canny reprend les matrices de Sobel, et applique quelques opérations supplémentaires pour obtenir des bords plus mieux définis. Les détails de l'algorithme sont décrits ci-dessous

1. **Réduction du bruit :**

On applique un filtre Gaussien 5x5 pour réduire le bruit présent dans l'image

2. **Recherche du gradient d'intensité de l'image :**

On applique ensuite un kernel de Sobel sur l'image "lisse" dans les directions verticales et horizontales afin d'obtenir les dérivées premières dans la direction verticale  $G_x$  et horizontales  $G_y$ . Ce procédé est identique à celui décrit plus haut.

3. **Suppression des non maximums locaux**

Une fois les gradients obtenus, on analyse tout les pixels de l'image, et on détermine si le pixel est un maximum local dans la direction du gradient.

Si oui, c'est un bord et sa valeur est gardée pour la prochaine étape,

---

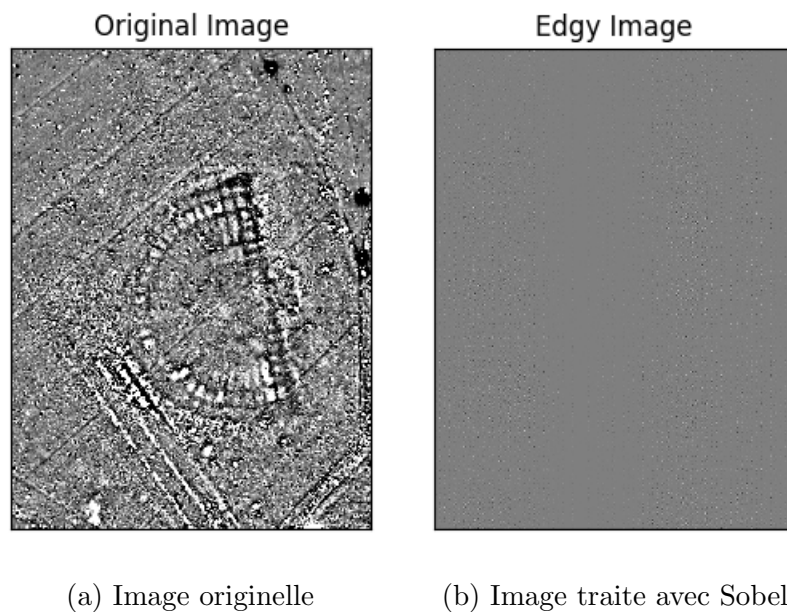


FIGURE 2 – Traitement du relevé magnetometrique de l'amphitheatre de Tarquimpol (Moselle), avec l'operateur Sobel

sinon, elle est mise a 0. On obtient une image binaire, avec que des bords

#### 4. Seuil d'Hysteresis

On utilise deux seuils,  $minVal$  et  $maxVal$ . Tout les bords ayant une intensité de gradient supérieur a  $maxVal$  est forcement un bord, ceux en dessous de  $minVal$  sont forcement des non-bords, et sont donc abandonne. Les bords qui sont entre ces deux seuil sont classe "bords" ou "non-bords" selon leur connectivité. Si ils sont connecte a des pixels qui sont des forcement des bords, alors ce sont des bords, sinon, ils sont aussi abandonne.

Encore une fois cet algorithme était déjà disponible dans la librairie d'analyse de d'image PythonCV2, nous n'avons pas eu a l'implémenter, et nous avons bénéficié de quelques optimisations rendant l'exécution plus rapide. Avec une image "propre", on obtient ce genre d'image :

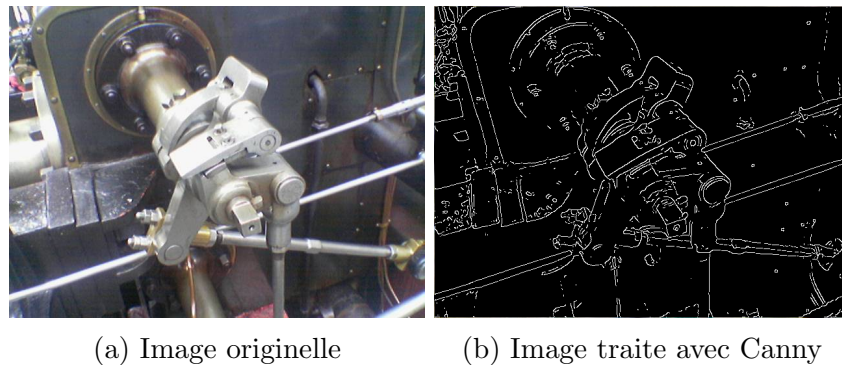
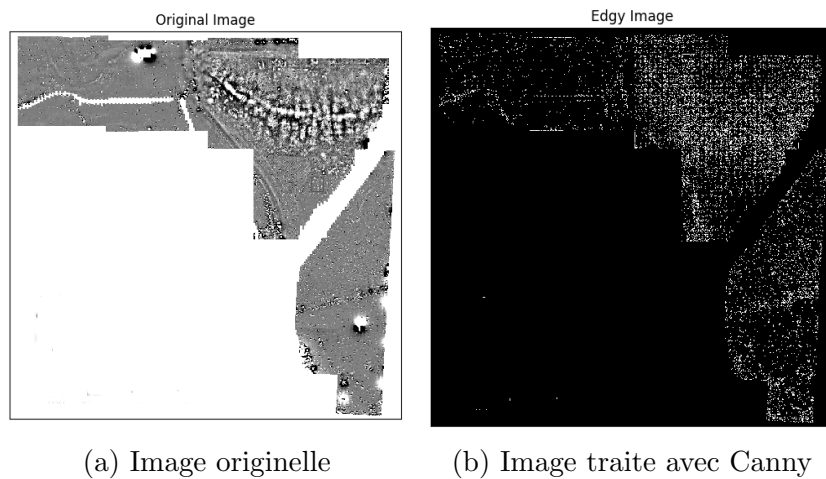


FIGURE 3 – Comparaison d'une image sans traitement avec celle traite avec Canny, de Wikipedia [5][4]

On observe que les bords sont bien plus défini, et propre par rapport a ceux obtenu avec l'opérateur Sobel. Mieux, les bords "majeurs", sont préservé, alors que les bords "mineurs" disparaissent. Cela est du aux deux dernières étapes de l'algorithme de Canny. Lorsque l'on applique Canny a nos images, nous obtenons ceci :



Même si il y a une amélioration certaine par rapport a Sobel, l'algorithme ne parvient toujours pas a trouver les contours des objets d'intérêt, encore une fois a cause du a la grande quantité de bruit que l'on trouve dans l'image. Cependant on observe une densité de points plus élevé aux niveau des endroits d'intérêts, comme ici, et une densité faible dans les zones ou on ne trouve ni objets ni bruit.

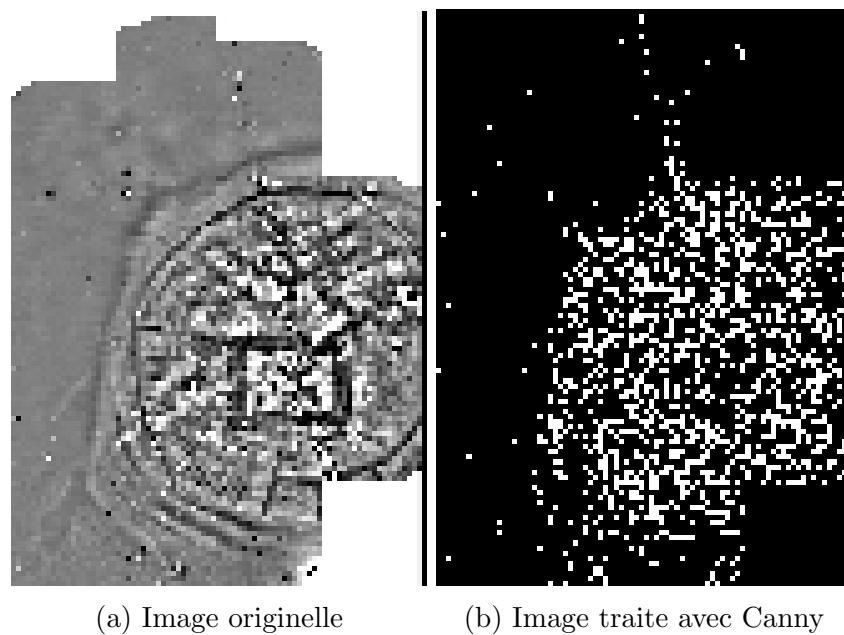


FIGURE 5 – Comparaison entre de détails entre une image sans traitement avec peu de bruit, et l'image traite avec Canny. Même si les bords ne sont pas détecté correctement, les lieux d'intérêt de l'image (la forme circulaire) ont très visiblement une haute densité de points.

### 5.2.3 Problèmes rencontrés

De toute évidence, nous avons atteint un obstacle : le bruit. Celui ci était présent sur toutes les images, au point d'empêcher toute tentative d'analyse classique de signal. La magnétométrie est très sensible aux objets métalliques, et dans la région où ont été fait les relevés, de nombreux objets métalliques contemporains sont présent dans le terrain, notamment à cause des combats de la 2<sup>ème</sup> guerre mondiale, mais aussi à cause des labours, et du déchets métallique produit par l'occupation humaines de ces terrains.

Une analyse classique ne suffisait donc pas, et le développement de méthodes de filtrage et de détection de bords étant capable de traiter des images aussi bruitées est hors de notre portée. Notre attention devait donc se porter sur la résolution d'un problème, annexe à celui qui nous intéressait, mais dont les techniques développées par sa résolution pouvait nous aider dans notre objectif principal.

## 6 Développement d'une nouvelle méthodologie

### 6.1 Détections d'objet métallique par réseaux neuronaux

Comme dit plus haut, les techniques classiques utilise dans l'analyse d'image sont inapplicables dans notre cas, du notamment au haut niveau de bruit présent dans les images a traites. Cependant, il est évidents que ce bruit n'est pas identique aux objets que nous cherchons, car sinon, les archéologues ne pourrait pas analyser ces images. Pour mieux diriger nos recherches nous avons choisi de détecter un type particulier de signaux, ceux provenant d'objet métalliques. Les reseaux de convolution sont tres adapte a cetttes taches ; car ils filtrent l'image et developpent eux meme les regles qui definissent l'objet ainsi que leur filtres. Ils permettent de detecter des motifs dans des parties de l'image, et non plus sur des pixels individuels, comme etait le cas avec le filtrage ou la detection de bords. De plus, nous construisons un reseau n'apprenant a detecter qu'une seule classe d'objet, ce qui rendra l'apprentissage plus rapide. Si nous reussissons a detecter ces objets métalliques, nous pouvons alors :

1. "Nettoyer" les cartes de ces objets, qui ont tendance a rendre difficile la lecture, en appliquant un flou gaussien par exemple
2. Utiliser le meme reseau dans la detection de plus d'objet, en transférant l'apprentissage deja fait sur un plus grand nombre de classe

#### 6.1.1 Caractérisation

Les dipôles proviennent des objets métalliques contemporains, et se caractérisent par un pôle positif très fort au niveau de l'objet, suivi d'un halo circulaire négatif, dont le diamètre est approximativement 2 fois celui de l'objet. On peut voir sur la figure 6 que les anomalies se ressemblent beaucoup, et son très présentes sur virtuellement toutes les cartes. De plus elles sont assez grandes pour être immédiatement reconnu sur nos relevés, ce qui simplifie la tache de labelisation des exemples a apprendre.

Un aspect interessant de ce probleme est, bien que si les dipoles produit par des objets métalliques contemporains sont generalement considere comme du bruit, ce n'est parce qu'ils *ne font pas partie de la periode recherche*. En d'autre terme, un objet métallique recent est un objet archeologique comme les autres : il provient d'une certaine periode historique, a une ou plusieurs techniques de fabrication qui lui sont propre, provient d'un certains lieu... Dans certaines circonstance de recherche, en particulier lorsque le site et les objets est date de la protohistoire, des objets datant de l'occupation romaine

---

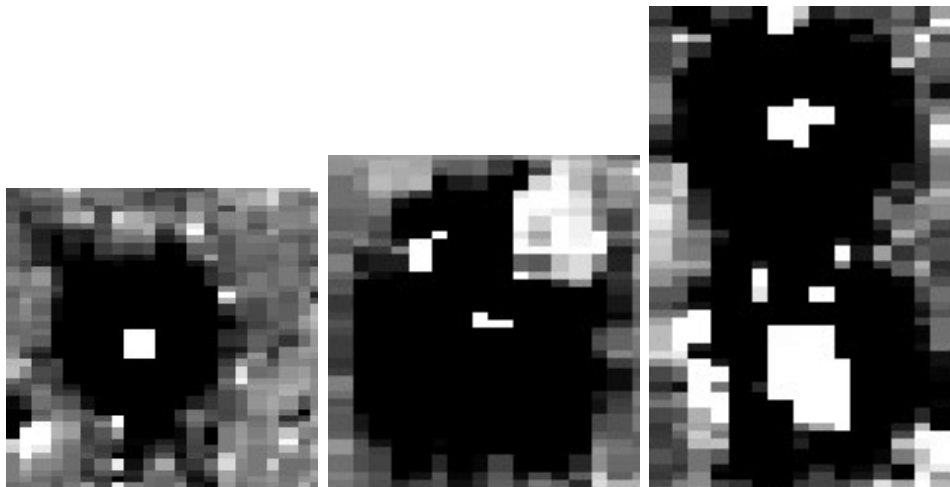


FIGURE 6 – Quelques exemples de dipôles, récupéré sur plusieurs relevés différents. On peut observer leur similitudes, tant par leur formes que par leur tailles

sont vu comme du bruit. C'est le cas avec le site de marsal, ou une occupation romaine a eu pour consequence des rejets de dechets datant de l'ere romaine, comme des tuiles, qui provoque du "bruit" sur nos cartes. En somme, il ne faut pas voir notre approche comme une recherche de "bruit", mais simplement comme un changement d'objet d'interet. Nous en choisissons un distinct, commun, et donc plus facile a etudier.

### 6.1.2 Explication de l'approche

Les réseaux de convolution sont une innovation récente dans le champs de l'apprentissage automatique. Le premier exemple d'un réseau de convolution moderne provient de Yann Le Cun [3] en 1998. En 2012, les réseaux de convolution réapparaîtront avec AlexNet [2], et restent aujourd'hui sur le devant de la scène de l'apprentissage automatique. Les réseaux de convolutions, que nous appellerons CNNs (Convolutionnal Neural Network) se composent de 2 parties

### 6.1.3 Problèmes initiaux

Un des difficulté principale attache a ce projet est le petit nombre initial d'exemples. En tout, nous possedons 41 Releve geophysique. De toute evidence, une telle quantite de donnee ne suffirait pas a entrainer un modele : les bases de donnees modernes, en particulier celle d'images se compose de plusieurs centaine de milliers d'exemples ; par exemple imageNet comporte

---

plus de 14 Millions d'exemple (<http://image-net.org/about-stats>). Nous devons alors trouver des methodes permettant d'augmenter artificiellement le nombre d'exemple en prenant garde que ces exemples cree artificielement represente toujours bien la realite. De plus nous ne possedions aucune competence en terme d'apprentissage automatique, et nous allions devoir tout apprendre.

#### 6.1.4 Solutions

##### Génération d'exemple :

Classifier a la main tout les exemples nécessaires a l'entraînement du réseau était une tâche surhumaine. Par soucis d'efficacité, nous avons décidé de construire nous même nos exemples, et des les utiliser en combinaison avec des exemples provenant des images, et classifie a la main. Pour avoir une meilleure chance de créer un bruit proche de la réalité, nous avons utilise 2 approches différentes :

- Une approche informatique
- Une approche mathématique

L'approche informatique consiste a créer 2 cercles, de tailles variables, un noir et un blanc, ayant la même origine. Ces cercles sont imprime sur une *backplate* grise avec du bruit génère aléatoirement. Afin d'améliorer le résultat, du bruit est ajoute aux cercles, augmentant particulièrement aux bords de ceux ci.

Nous avons génère une fonction mathématique se rapprochant de celle d'un dipôle.

Il s'agit d'une fonction sinusoïdale amortie exponentiellement modifie pour que son domaine soit  $\mathbb{R}^2 \rightarrow \mathbb{R}$  :

$$f(x, y) = A.e^{-\lambda.\sqrt{(x+i)^2+(y+j)^2}}.\cos(\omega.\sqrt{(x+i)^2+(y+j)^2} + \phi)$$

Avec :

$A$  l'amplitude initiale, choisie aléatoirement entre 200 et 255

$\lambda$  coefficient d'amortissement

$\omega$  vitesse angulaire

$\phi$  angle de phase a l'origine

$i, j$  décalage horizontal et vertical, respectivement, par rapport a l'origine

Avec la génération aléatoire des paramètres de cette fonction, nous pouvons obtenir une grande variété d'exemple qui ressemble aux dipôles retrouvés dans nos cartes. **Traitement des images :**

Afin d'obtenir une meilleure qualité d'exemples nous avons utilise ceux provenant de nos données originelle. Nous avons d'abord découpe chacune des images en carre de taille 96\*96 pixels ; en décalant le carre de 48 pixels



verticalement et horizontalement a chaque fois, nous pouvons obtenir encore plus d'exemples. Au final, nous avons pu obtenir plus de 51000 carres, grâce au fait que les images étaient très grande (souvent plus de 2000\*2000 pixels). Nous avons ensuite, à la main, trouvé tous les carres contenant un dipôle, ce qui nous donne 413 images de dipôles. Ces images, provenant du monde réel sont très variées, que ce soit par leur taille, leur position dans l'image ou encore leur nombre. // Cependant, 400 images n'est toujours pas suffisant pour entraîner un réseau. Pour enrichir notre dataset nous avons décidé d'appliquer quelques transformations sur chacune des images. À l'heure où nous écrivons ces lignes, nous avons implémenté les transformations suivantes :

1. Rotation de l'image sur l'axe x
2. Rotation de l'image sur l'axe y
3. Ajout de bruit

### Implémentation Réseau Neuronal

Nous avons implémenté un réseau neuronal classique s'entraînant sur des exemples générés. Le réseau reprends l'architecture de l'exemple de TensorFlow pour la classification de chiffres manuscrit avec une modification du nombre d'entrée pour s'adapter à la taille des exemples générés. Lors de l'entraînement, on a pu observer une très rapide maximisation du taux de détection, arrivant à un taux de réussite de 100% sur les exemples de test en 4 époques. Nous avons ensuite tenté d'appliquer ce réseau sur une carte, dans l'espoir de créer une *heatmap* d'activation du réseau. Malheureusement, ce réseau possédait un taux d'erreur très élevé sur les exemples réels. Nous ne possédions pas le temps d'implémenter un réseau neuronal de convolution, et les résultats obtenus avec le réseau neuronal "classique" nous indiquaient qu'utiliser seul des exemples générés ne suffirait pas pour obtenir un bon taux de détection sur les données réelles. Nous avons donc préféré utiliser notre temps restant à construire une meilleure base d'exemple en mixant données réelle et données générée, pour obtenir non seulement une meilleure variance, mais un *dataset* plus proche de la réalité.

---

## 7 Bilan provisoire et perspectives

### 7.1 Conclusion

Ce semestre fut riche en travail et en résultat. Nous avons énormément appris sur le travail d'analyse de données, le nettoyage de *dataset*, les réseaux d'apprentissage automatique, ainsi que des techniques d'analyse d'image. Le travail de mise en place de système d'apprentissage automatique est un travail long et délicat ; d'autant plus que nous possédons un corpus de données réduit. Malgré tout nous, nous sommes optimiste sur le déroulement de ce projet. Le travail de ce premier semestre ne fut pas que lecture d'article académique : nous avons pu observer que les techniques traditionnelles d'analyse d'image ne fonctionnent pas avec nos données, nous avons créé un corpus de données annoté de plus de 50 000 images, et pose les bases pour l'implémentation d'un réseau de convolution.

### 7.2 Implémentation d'un réseau de convolution

La prochaine étape est évidemment l'implémentation d'un réseau de convolution, apprenant sur le *dataset*. La partie la plus chronophage, celle de préparer la base de données étant déjà accompli, cette tâche ne prendra que peu de temps.

### 7.3 Choix de la direction de recherche

Nous devons maintenant décider de la direction de nos recherches pour le Semestre 2. L'implémentation d'un réseau de convolution, et l'analyse des résultats obtenus, nous permettent de faire une décision sur le choix des objets à analyser. Si le réseau donne une précision en deca de nos attentes, nous continuerons notre travail sur l'analyse d'objets métalliques. Si nous obtenons une précision acceptable alors nous pouvons alors étendre celui-ci à l'étude d'autres objets archéologiques dont la définition en tant que structure dans une image est plus complexe, comme des tombes, des routes, etc.

Nous désirons également pouvoir travailler avec d'autres types de relevés, à utiliser en conjonction des relevés magnétométrique, comme par exemple le LIDAR, ou le radar de sol. Cependant obtenir ces relevés n'est pas une tâche aisée, d'autant plus qu'ils couvrent généralement des zones différentes de celles couvertes par la magnétométrie.

Ensuite, nous voudrions encore enrichir notre corpus de données : des avancées récentes ont été faites à ce sujet [DatasetTranfo], et il serait intéressant de pouvoir les utiliser. Finalement, nous nous posons la question de

---

rendre ce corpus publique, apres autorisation aupres des proprietaires, bien sur. Des corpus sur les objets archeologiques dans ce contexte sont virtuellement inexistant, et serais un ajout interessant.

## 8 Annexes

## Références

- [1] J. CANNY. “A Computational Approach to Edge Detection”. In : *IEEE Transactions on Pattern Analysis and Machine Intelligence* PAMI-8.6 (nov. 1986), p. 679–698. ISSN : 0162-8828. DOI : 10.1109/TPAMI.1986.4767851.
- [2] Alex KRIZHEVSKY, Ilya SUTSKEVER et Geoffrey E HINTON. “ImageNet Classification with Deep Convolutional Neural Networks”. In : (2012). Sous la dir. de F. PEREIRA et al., p. 1097–1105. URL : <http://papers.nips.cc/paper/4824-imagenet-classification-with-deep-convolutional-neural-networks.pdf>.
- [3] Y. LECUN et al. “Gradient-Based Learning Applied to Document Recognition”. In : (), p. 306–351.
- [4] SIMPSON'S CONTRIBUTOR. *Valve monochrome canny (6)*. 2008. URL : [https://commons.wikimedia.org/wiki/File:Valve\\_monochrome\\_canny\\_\(6\).PNG#/media/File:Valve\\_monochrome\\_canny\\_\(6\).PNG](https://commons.wikimedia.org/wiki/File:Valve_monochrome_canny_(6).PNG#/media/File:Valve_monochrome_canny_(6).PNG).
- [5] SIMPSON'S CONTRIBUTOR. *Valve Original (1)*. 2008. URL : [https://commons.wikimedia.org/wiki/File:Valve\\_original\\_\(1\).PNG](https://commons.wikimedia.org/wiki/File:Valve_original_(1).PNG).
- [6] SIMPSON'S CONTRIBUTOR. *Valve sobel (3)*. 2008. URL : [https://commons.wikimedia.org/wiki/File%3AValve\\_sobel\\_\(3\).PNG](https://commons.wikimedia.org/wiki/File%3AValve_sobel_(3).PNG).
- [7] Irwin SOBEL. “An Isotropic 3 3 Image Gradient Operator”. In : (fév. 2014).