Ingmar Gbiorczyk-Morel
Antoine Thibaut

ESILV A5 IBO

Rapport Projet PAS : Détection de tableau

ESILV LÉONARD DE VINCI | ÉCOLE D'INGÉNIEURS PARIS-LA DÉFENSE

12/01/2020

Sommaire

Introduction

La Reconnaissance Optique de Caractère (OCR) ou Optical Character Recognition correspond aux différents procédés informatiques utilisés pour la traduction d'images comportant des textes imprimés (tableaux excel, image de facture, scanner d'une carte d'identité …) afin de ressortir ces informations en fichier textes ou csv.

Ces outils sont de plus en plus utilisés de nos jours afin d'automatiser certaines tâches, comme la gestion des factures ou la vérification des données d'identités et bancaires.
L'utilisation d'un logiciel d'OCR permet donc de récupérer les textes présents dans l'image et de les sauvegarder dans un fichier afin de pouvoir les traiter par la suite ou de les stocker dans une base de données.

Les premiers systèmes d'OCR réalisés avaient besoin d'une longue phase d'apprentissage afin de comprendre et interpréter les caractères données. A présent, les systèmes sont plus intelligents et peuvent reconnaître la plupart des des polices ou caractères avec un niveau de précision presque parfait.

Il y a encore quelques années les systèmes d'OCR performants étaient protégés et les logiciels disponibles étaient open-source et donc développés par des amateurs, ce qui signifie qu'ils n'étaient pas vraiment performants exemple de GOcr. Cependant depuis 2006 et l'apparition de Tesseract (dont nous parlerons plus bas) qui est un système open-source assez performant a changer la situation.

Nous allons maintenant parler du fonctionnement de ces systèmes. Comme dit précédemment, l'OCR part d'une image numérique réalisée par un scanner ou un appareil photo afin de sortir un fichier texte, cependant certains logiciels vont encore plus loin et vont même jusqu'à retranscrire les informations du texte (police, gras, italique ..) ainsi que les différentes mise en page appliqués au document, mais peuvent également reconstruire les tableaux.

Les différentes étapes sont :

- La préanalyse : Durant cette phase, le but est d'analyser l'image afin d'améliorer la qualité de celle-ci si cela est possible. Cela signifie, le redressement de l'image, la changement du contraste, le passage en niveau de gris, en noir et blanc, afin d'optimiser au maximum l'image pour obtenir une sortie parfaite.

- La segmentation : Cette phase à pour but d'isoler des blocks de texte à l'intérieur de l'image ce qui va permettre également d'améliorer la détection.

- La reconnaissance : La reconnaissance des caractères présents. Elle va comparer avec une bibliothèque de formes connues et ainsi ressortir la forme la plus proche de celle trouvée. Il existe différentes techniques de reconnaissance (Classification, méthodes métriques et méthodes statistiques).

- Le post-traitement : Utilisation de méthodes linguistiques (langues) afin de minimiser le nombre d'erreurs lors de la reconnaissance. Ce système utilise des règles basées sur des dictionnaires de mots, des syllabes … Il est également possible d'utiliser des bases de données afin d'éliminer les solutions incorrects (comme par exemple un dictionnaire de prénom, ou une base de données avec les noms de rues …)

- La sortie : Création du fichier de sortie (texte, csv) sans mise en page, la mise en page est présente pour les systèmes les plus performants et les plus développés.

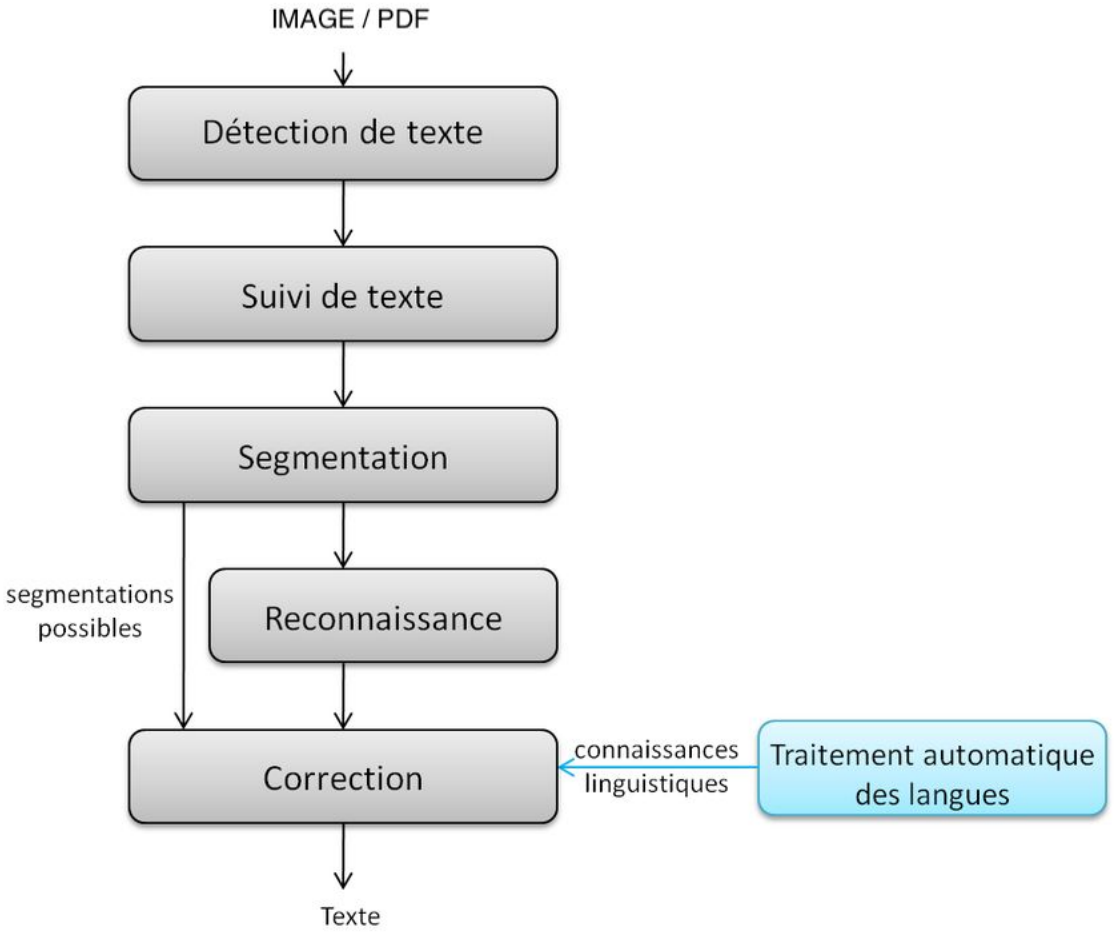


IMAGE / PDF
Détection de texte
Suivi de texte
Segmentation
segmentations possibles
Reconnaissance
Correction
connaissances linguistiques
Traitement automatique des langues
Texte

Bibliographie

- Fiche d'emploie sur un poste concernant la détection de tableau dans les documents
- Document concernant la détection de tableau dans des documents complexes
- Archive concernant l'utilisation de couleurs pour l'extraction de tableau dans des documents
- Informations concernant le projet MORDOR
- Rapprochement des données pour la reconnaissance d'entités dans les documents océrisés
- Extraction de donnée tabulaire dans des pdfs
- Tutoriel Google Vision sur la détection de document
- Documentation sur l'ocr de microsoft
- Github du projet Tesseract

Présentation de la problématique

Le but principal de ce projet est donc d'utiliser un logiciel de reconnaissance optique de caractères sur des images comportant des tableaux et donc des fichiers comme excel, pdf ou des photos de factures afin d'en extraire les données présentes et les ressortir dans un fichier texte ou un fichier CSV.

La complexité de ce problème est donc de réussir à ressortir les différentes informations tout en gardant l'ordre du tableau et réussir à bien remettre chaque information dans la bonne ligne et la bonne colonne afin d'éviter toute erreur par la suite. Mais il est également nécessaire de récupérer les bonne informations et que la compréhension des caractères des outils utilisés soit tout de même assez évolué et pertinente.

Pour réaliser cela nous allons devoir chercher les différents logiciels disponibles, comprendre leurs fonctionnements ainsi que voir comment les adapter à notre projet.

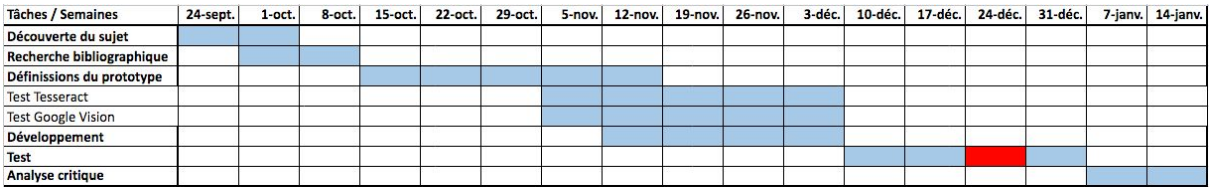
Il est donc nécessaire pour ce faire d'avoir divers types de tableaux (assez variés) afin d'optimiser la détection (des tableaux excels, des tableaux pdf, des photos de factures, des tableaux réalisés à la main ..). Les tableaux peuvent être d'origines multiples il est donc nécessaire d'avoir une gestion similaire avec un traitement intermédiaire afin de standardiser cela.

Pour ce faire nous avons donc choisis d'utiliser deux différentes API : Google Vision et Tesseract, dont nous allons parler par la suite.

Organisation

Nous avons réalisé ce projet à deux. Pour ce faire nous avons dû nous organiser sur les différentes tâches à réaliser afin d'optimiser le travail et ne pas perdre de temps. En effet, le temps pour réaliser ce projet, et avec la charge de travail à réaliser se révélant assez court, une bonne organisation était primordiale.

De ce fait, au début du projet nous avons fait un planning des tâches à réaliser ou nous avons quantifier la charge de travail pour chaque partie.



Tâches / Semaines	24-sept.	1-oct.	8-oct.	15-oct.	22-oct.	29-oct.	5-nov.	12-nov.	19-nov.	26-nov.	3-déc.	10-déc.	17-déc.	24-déc.	31-déc.	7-janv.	14-janv.
Découverte du sujet																	
Recherche bibliographique																	
Définissions du prototype																	
Test Tesseract																	
Test Google Vision																	
Développement																	
Test																	
Analyse critique																	

Nous allons donc à présent expliquer chaque partie :

- Découverte du sujet : Cela concerne les premières recherches réalisées concernant les OCRs, les différents outils déjà présents et le début de l'organisation sur le projet.

- Recherche bibliographique : Recherches dans différents ouvrages de référence concernant le sujet afin d'obtenir des informations sur les OCRs ainsi que leurs fonctionnement.

- Définition du prototype : Définition des différentes qualités techniques et des caractéristiques du fonctionnement du produit, afin de savoir ce que nous devons implémenter.

- Test Tesseract : Prise en main de l'outil Tesseract de google et réalisation de premiers tests sur des tableaux afin de comprendre le fonctionnement et le traitement à réaliser.

- Test Google Vision : Prise en main de l'outil Google Vision et réalisation de plusieurs tests sur différents tableaux, afin de comprendre le fonctionnement et les retours.

- Développement : Réalisation de scripts permettant de traiter les données récupérées par les deux OCRs et de les ressortir.

- Test : Amélioration des scripts et correction des différentes erreurs rencontrées durant le développement des scripts

- Analyse Critique : Analyse du projet dans sa globalité, des erreurs réalisées ..

Etat de l’art

Avant de commencer nous avons donc fait un etat de l’art des logiciels et solutions qui existait afin de pouvoir travailler dessus et les comprendre. Nous allons donc les présenter ci-dessous :

- ## API Azure Vision :

  Il y a deux API, une pour lire et une pour écrire (les deux utilisent la technologie OCR).
  L’api qui nous intéresse ici est celle pour lire. Celle-ci est utilisable pour différents types de fichiers (JPEG, PNG, BMP, PDF et TIFF) dans plusieurs angles, avec plusieurs langues, manuscrites ou non, image ou texte. Les résultats sont envoyés sous format JSON.
  Cette solution est payante.

- ## API Google Vision :

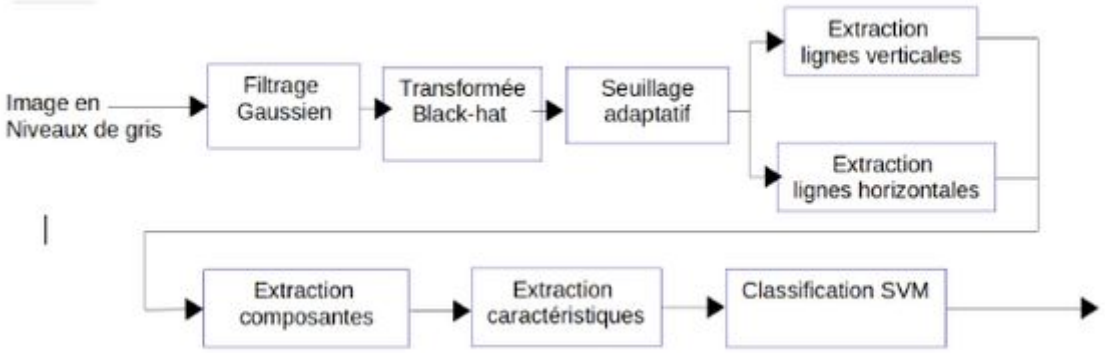
  Cette API est assez similaire à celle de Azure, elle semble cependant mieux documentée, avec de nombreux tutoriels en Python et documentations.
  Elle supporte également plusieurs types de fichiers (JPEG, PNG, BMP, PDF et TIFF). Le format de réponse est en JSON aussi, mais elle possède une connexion simple avec d’autres outils de Google Cloud (comme Translate notamment).
  Cette solution est payante.

- ## API Tesseract :

  Cette API est open-source et a été développée par google. Elle est assez bien documentée et utilisable en python (pytesseract). Elle supporte différents types de fichiers (JPEG, PNG, BMP et TIFF). Le retour peut se faire en format txt ou csv.

- Projet MAURDOR :

  La direction générale de l'armement a notamment lancé un projet MAURDOR sur le sujet, publié en 2013.
  Une recherche de 2014 de 2 universités contribue à ce projet MAURDOR avance un nouveau prototype utilisant un classifieur linéaire,se concentrant sur les contours pour définir si un élément appartient ou non au tableau (sans réellement traitement de la donnée.



Présentation du jeu d’essai

Afin de tester Google Vision ainsi que Tesseract nous avons réalisé un jeu d’essai comprenant différents types de documents dans le but de pouvoir généraliser ce traitement à des images, des factures, des tableaux fait à la main, des pdfs …

Pour ce faire nous avons donc différents type de tableaux :

- Des tableaux pdfs avec des données et des couleurs :

SOCIETE		déc-14											
DATE (date figurant sur la facture)	CODE JOURNAL (journal des achats)	N° DE COMPTE (compte fournisseur)	N° DE PIECE (N° de la facture)	LIBELLE	DEBIT	CREDIT	M (euros)	60608000-FOURNITURES DE BUREAU	60611000-ELECTRICITE	61320000-LOCATIONS IMMOBILIERES	62600000-FRAIS POSTAUX	62610000-TELEPHONE + INTERNET	TVA 20%
03/12/2014	HA	401POSTE	12345678910	La Poste		8,5	E				8,5		
03/12/2014	HA	401ORANGE	9876543210	Orange		55	E					45,83	9,17
	HA	401				0	E						
	HA	401				0	E						
	HA	401				0	E						
	HA	401				0	E						
								0	0	0	8,5	45,83	9,17
31/12/2014	HA	44566000		TVA DEDUCTIBLE SUR ACHATS	9,17		E						
31/12/2014	HA	60608000		FOURNITURES DE BUREAU	0		E						63,5
31/12/2014	HA	60611000		ELECTRICITE	0		E						
31/12/2014	HA	61320000		LOCATIONS IMMOBILIERES	0		E						
31/12/2014	HA	62600000		FRAIS POSTAUX	8,5		E						
31/12/2014	HA	62610000		TELEPHONE + INTERNET	45,83		E						
					63,5	63,5							

Tableau de bord résumé de la Brasserie des sports

	n		n-1		
	€	%	€	%	
Chiffre d'affaires	742 000	100	698 000	100	
Nombre de couverts	*45 800*		*44 200*		
Addition moyenne	*16,20*		*15,79*		
- consommations de matières	214 100	28,85	205 600	29,46	(1)
= marge brute	527 900	71,15	492 400	70,54	
- charges de personnel (masse salariale)	275 000	37,06	270 000	38,68	(2)
= marge sur coût principal	252 900	34,08	222 400	31,86	
- Frais généraux	88 000	11,86	102 200	14,64	(3)
= résultat brut d'exploitation	164 900	22,22	120 200	17,22	
- coûts d'occupation	106 750	14,39	101 400	14,53	(4)
= RÉSULTAT COURANT	58 150	7,84	18 800	2,69	

- Des photos de factures :

Berghotel
Grosse Scheidegg
3818 Grindelwald
Familie R.Müller

Rech.Nr. 4572 30.07.2007/13:29:17
Bar Tisch 7/01

2xLatte Macchiato	à	4.50	CHF	9.00
1xGloki	à	5.00	CHF	5.00
1xSchweinschnitzel	à	22.00	CHF	22.00
1xChässpätzli	à	18.50	CHF	18.50

Total : CHF **54.50**

Incl. 7.6% MwSt 54.50 CHF: 3.85

Entspricht in Euro 36.33 EUR
Es bediente Sie: Ursula

MwSt Nr.: 430 234
Tel.: 033 853 67 16
Fax.: 033 853 67 19
E-mail: grossescheidegg@bluewin.ch

REI

222 Yale Avenue North
Seattle, WA 98109
(206) 223-1944

Cashier 130957 Register 35 Store 11
SALE Tran# 3498 12/31/14 5:59 PM

02001135349812312014415418513O

MEMBER # 15418513

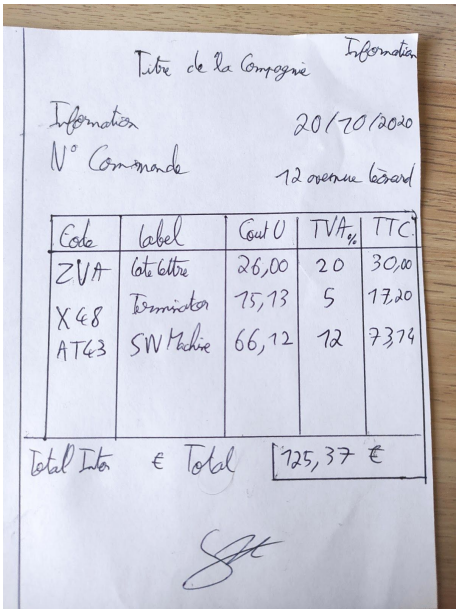
858-549-0001	Barracuda	299.83 *
886888979771	Cartel EST	269.95
96	Belt Wax-96	10.00 *
164	SB Mount New-164	15.00 *
1	Membership	20.00 *N

SUBTOTAL		614.78
SALES TAX	9.50%	56.50
TOTAL		671.28
BANKCARD		671.28

ACCT # ***********3017
AUTH # 521743

* Indicates Non-Dividendable Item

- Des tableaux réalisés à la main :



Titre de la Compagnie Information

Information 20/10/2020
N° Commande 12 avenue Gérard

Code	label	Cout U	TVA %	TTC
ZVA	Cote Cotte	26,00	20	30,00
X48	Terminator	15,13	5	17,20
AT43	SW Machine	66,12	12	73,74

Total Info € Total 725,37 €

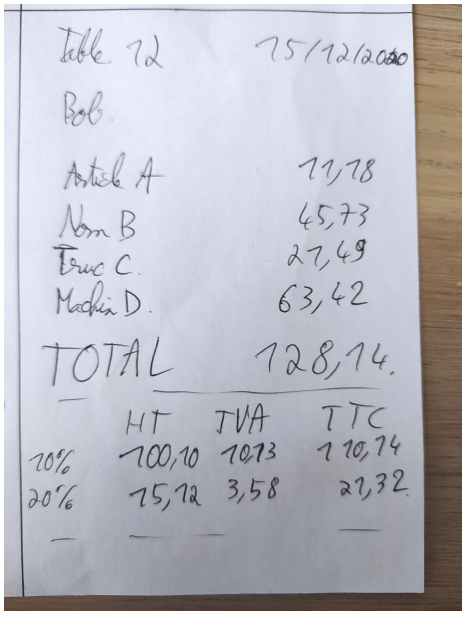


Table 12 15/12/2020
Bob

Article A	11,78
Nom B	45,73
Truc C.	27,49
Machin D.	63,42
TOTAL	128,14.

	HT	TVA	TTC
10%	100,10	10,13	110,14
20%	15,12	3,58	21,32

- Des tableaux réalisés avec excel :

Sheet1

Année	Prix achat marchandise	Frais	Chiffre d'affaire	Bénéfices	Loyer
2015	2 300,00 €	127,00 €	18 930,00 €	16 503,00 €	912,00 €
2016	2 400,00 €	348,00 €	19 530,00 €	16 782,00 €	912,00 €
2017	2 430,00 €	110,00 €	21 930,00 €	19 390,00 €	912,00 €
2018	2 500,00 €	560,00 €	23 830,00 €	20 770,00 €	912,00 €
2019	2 840,00 €	430,00 €	26 730,00 €	23 460,00 €	912,00 €
2020	3 000,00 €	999,00 €	29 990,00 €	25 991,00 €	912,00 €

Sheet1

Année	Prix achat marchandise	Frais	Chiffre d'affaire	Bénéfices	Loyer
2015	2 300,00 €	127,00 €	18 930,00 €	16 503,00 €	912,00 €
2016	2 400,00 €	348,00 €	19 530,00 €	16 782,00 €	912,00 €
2017	2 430,00 €	110,00 €	21 930,00 €	19 390,00 €	912,00 €
2018	2 500,00 €	560,00 €	23 830,00 €	20 770,00 €	912,00 €
2019	2 840,00 €	430,00 €	26 730,00 €	23 460,00 €	912,00 €
2020	3 000,00 €	999,00 €	29 990,00 €	25 991,00 €	912,00 €

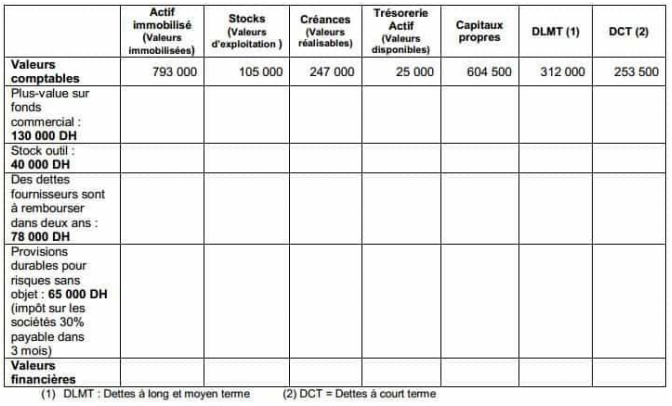
Sheet1

Année	Prix achat marchandise	Frais	Chiffre d'affaire	Bénéfices	Loyer
2015	2 300,00 €	127,00 €	18 930,00 €	16 503,00 €	912,00 €
2016	2 400,00 €	348,00 €	19 530,00 €	16 782,00 €	912,00 €
2017	2 430,00 €	110,00 €	21 930,00 €	19 390,00 €	912,00 €
2018	2 500,00 €	560,00 €	23 830,00 €	20 770,00 €	912,00 €
2019	2 840,00 €	430,00 €	26 730,00 €	23 460,00 €	912,00 €
2020	3 000,00 €	999,00 €	29 990,00 €	25 991,00 €	912,00 €

Travaux Tesseract

Comme dit précédemment, Tesseract possède un moteur OCR et un programme qui s'utilise directement en ligne de commande " Tesseract" ou "Pytesseract" avec du python. Nous avons utilisé Pytesseract 4 qui possède un moteur OCR basé sur un réseau de neurone (LTSM) qui est basé sur la reconnaissance de ligne, mais possède toujours le système de Tesseract 3 qui fonctionne à l'aide de la reconnaissance des modèles de caractères. Il possède également une gestion des langues qui permet de retrouver les mots plus facilement.

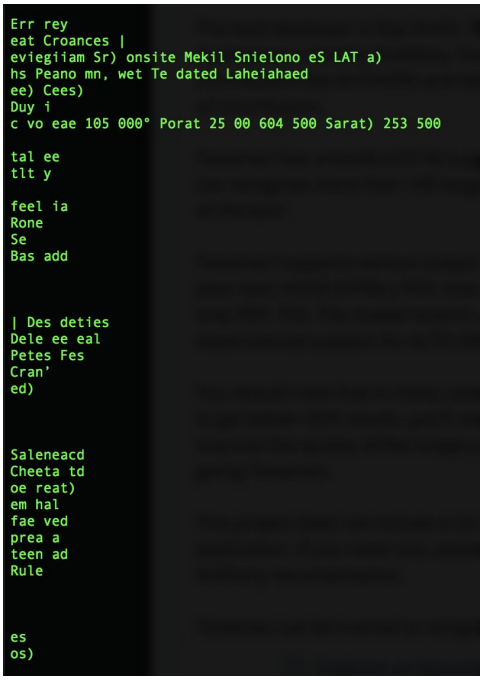
Dans un premier temps nous avons essayé d'utiliser Tesseract directement sur notre jeu d'essai afin de voir si la forme du résultat obtenu :



	Actif immobilisé (Valeurs immobilisées)	Stocks (Valeurs d'exploitation)	Créances (Valeurs réalisables)	Trésorerie Actif (Valeurs disponibles)	Capitaux propres	DLMT (1)	DCT (2)
Valeurs comptables	793 000	105 000	247 000	25 000	604 500	312 000	253 500
Plus-value sur fonds commercial : **130 000 DH**							
Stock outil : **40 000 DH**							
Des dettes fournisseurs sont à rembourser dans deux ans : **78 000 DH**							
Provisions durables pour risques sans objet : **65 000 DH** (impôt sur les sociétés 30% payable dans 3 mois)							
Valeurs financières							

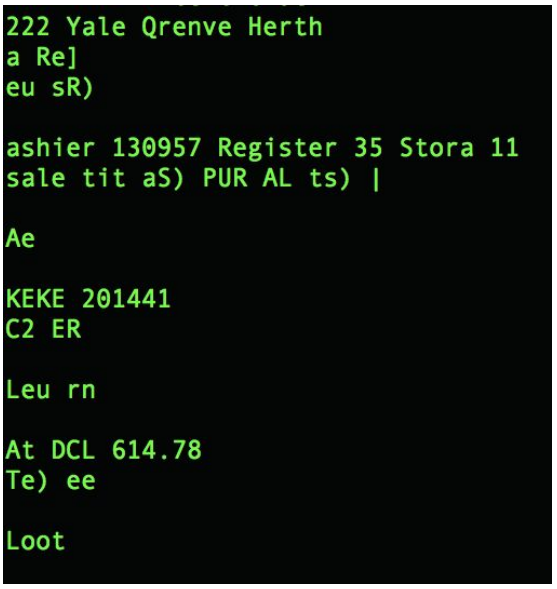
(1) DLMT : Dettes à long et moyen terme (2) DCT = Dettes à court terme

Tableau en entré



```
Err rey
eat Croances |
eviegiiam Sr) onsite Mekil Snielono eS LAT a)
hs Peano mn, wet Te dated Laheiahaed
ee) Cees)
Duy i
c vo eae 105 000° Porat 25 00 604 500 Sarat) 253 500

tal ee
tlt y

feel ia
Rone
Se
Bas add



| Des deties
Dele ee eal
Petes Fes
Cran'
ed)



Saleneacd
Cheeta td
oe reat)
em hal
fae ved
prea a
teen ad
Rule



es
os)
```

Sortie Tesseract



Facture en entrée ***Sortie Tesseract***

Nous pouvons voir que nous arrivons déjà à obtenir quelques retours, et nous arrivons à extraire certaines données. Cependant la compréhension des mots n'est pas correcte, il y a énormément d'erreurs et certains documents restent illisibles.

Afin d'améliorer cela nous allons effectuer un pré-traitement sur les images afin d'améliorer la compréhension de Tesseract. Pour ce faire 4 fonctions ont étés implémentées :

- Higher_reso : Permet d'améliorer la résolution de l'image
- Only_black : Passe l'image en noir et blanc
- Erosion : Permet de réduire l'érosion sur l'image tel que les zones d'ombres
- AdaptiveThreshold : Méthode de seuillage

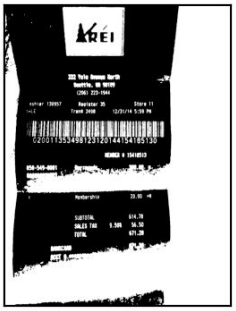
Afin de voir l'évolution de l'image nous avons afficher chaque image après le passage de chaque fonction et l'affiche du texte qu'elle contient :

```
img = cv2.imread(img_str)
img_rescale = rescal_img(img)

higher_reso2 = cv2.pyrUp(img_rescale)
if DEBUG:
    titles.append('higher_reso2'), images.append(higher_reso2)



only_black = cv2.inRange(higher_reso2, np.array([0,0,0]), np.array([150,150,150]))
if DEBUG:
    titles.append('only_black'), images.append(only_black)


img_erosion = cv2.erode(only_black, kernel, iterations=1)
if DEBUG:
    titles.append('img_erosion'), images.append(img_erosion)

ret, img_grey = cv2.threshold(img_erosion, 127, 255, cv2.THRESH_BINARY_INV)

img_bn = cv2.adaptiveThreshold(img_grey,255,cv2.ADAPTIVE_THRESH_GAUSSIAN_C, cv2.THRESH_BINARY,11,2)
if DEBUG:
    titles.append('img_bn'), images.append(img_bn)

img_a = cv2.adaptiveThreshold(img_grey,255,cv2.ADAPTIVE_THRESH_MEAN_C,cv2.THRESH_BINARY,11,2)
```

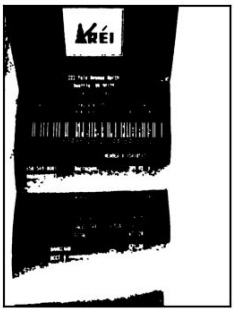
Script pour le pré-traitement des images

higher_reso2

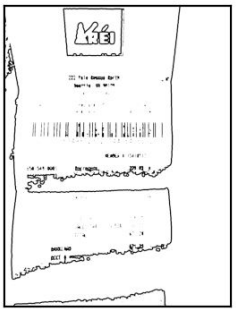
only_black



img_erosion



img_bn



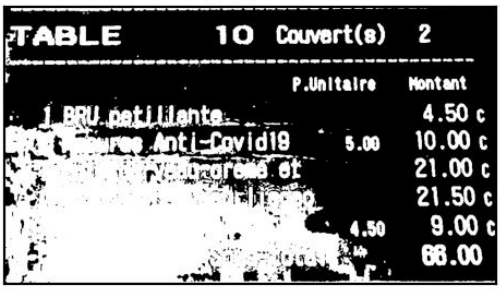
Résultat de l'image après traitement

```
Mii

222 Yale Avenue North
Seattle, Ua 98109



(206) 223-1944
: $7 —_astatar 35 tore
Tront 3498 12/31/18 5:59 PA
i WH |
WANT) AN I
011353498123 :
EWBER #15418
58-549-0001 Barracuda 230.83 +
86800979771 Cartel EST 269.35
% Beit Ver-96 10.00 +
164 SB Mount Nev-164 15.00 *
eae 70.00 #4



9.508 56.50
TOTAL 1.28

BANKCARD ons
acct # xesecess10939017

AUTH & 521743,

a ta je
```

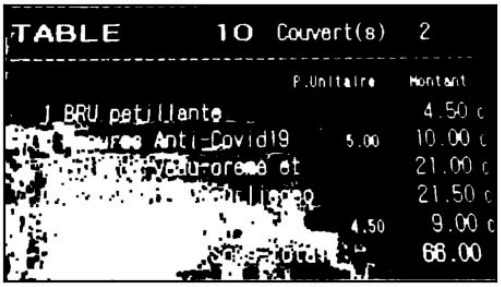
Résultat de sortie Tesseract

higher_reso2

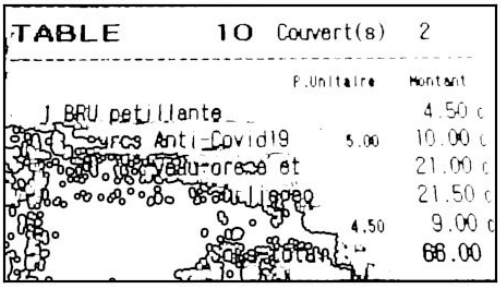
only_black



img_erosion



img_bn



Résultat de l'image après traitement

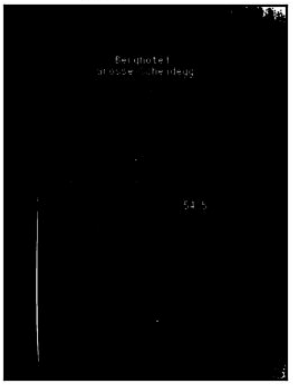
```
P.Unitaire Montant
1 BRU Peat e 4.50¢
oie de veau creme e 21.00 ¢
1 Rognons de veau | iegeo 21.50 ¢
2 Café

450 9.00¢
Sous-total : 66.00
```

Résultat de sortie Tesseract

higher_reso2

only_black

img_erosion



img_bn

Résultat de l'image après traitement

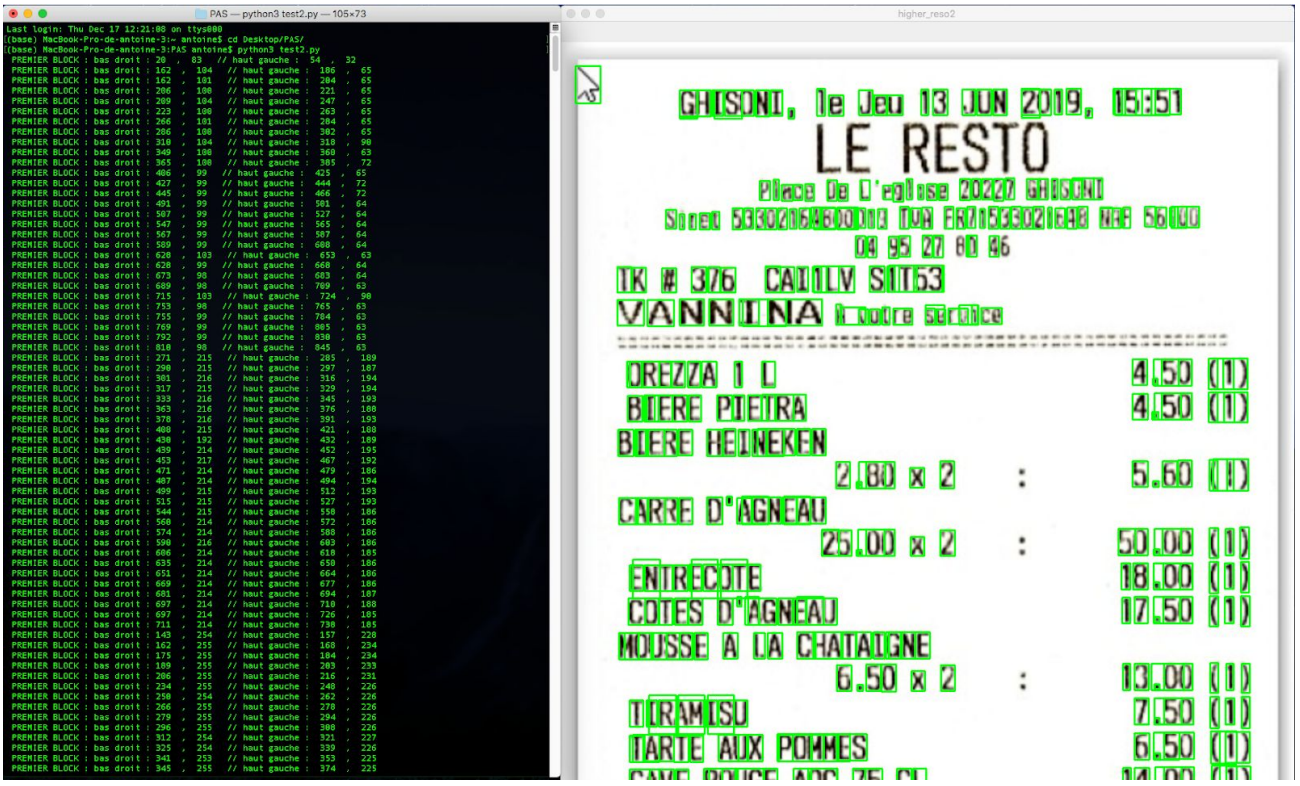
```
Berghotel
Grosse Scheidegg
3818 Grindelwald
Familie R.MGl ler

Rech. Nr. 4572 30. 07. 2007/ 13:29:17
Bar Tisch 7/01

exLatte Macchiato a 4.50 CHF 9.00
AxGloki @ 5.00 CHF 5.00
IxSchweinschnitzel a 22.00 CHF 22.00
IxChasspatz 11 @ 18.50 CHF 18.50

Total: CHF

Incl. 7.6% MwSt 54.50 CHF: 3.85

Entspricht in Euro 36.33 EUR
Es bediente Sie: Ursula

MwSt Ne. : 430 234
Tel.: 033 853 67 16
Fax. : 033 853 67 19
E-mail: grossescheidegg@bluewin. ch
```

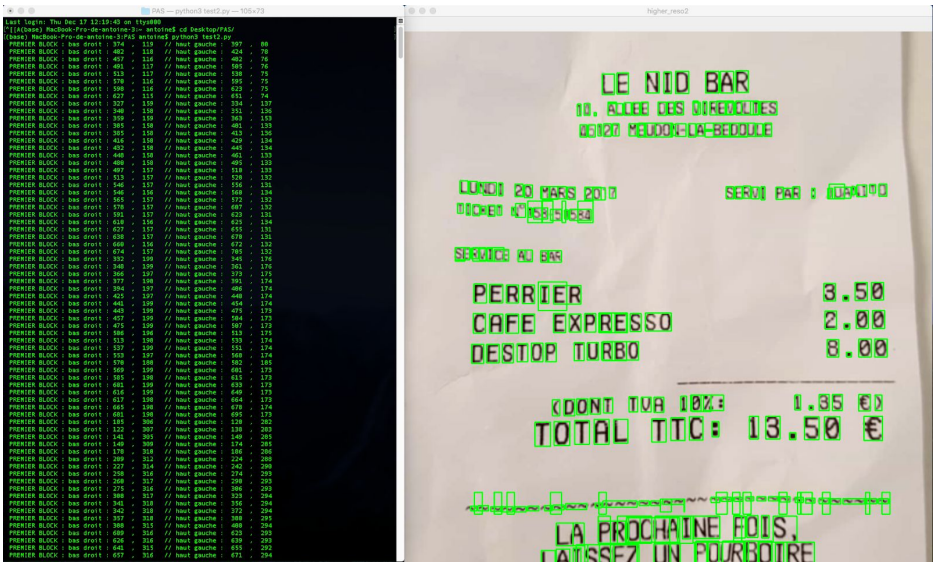
Résultat de sortie Tesseract

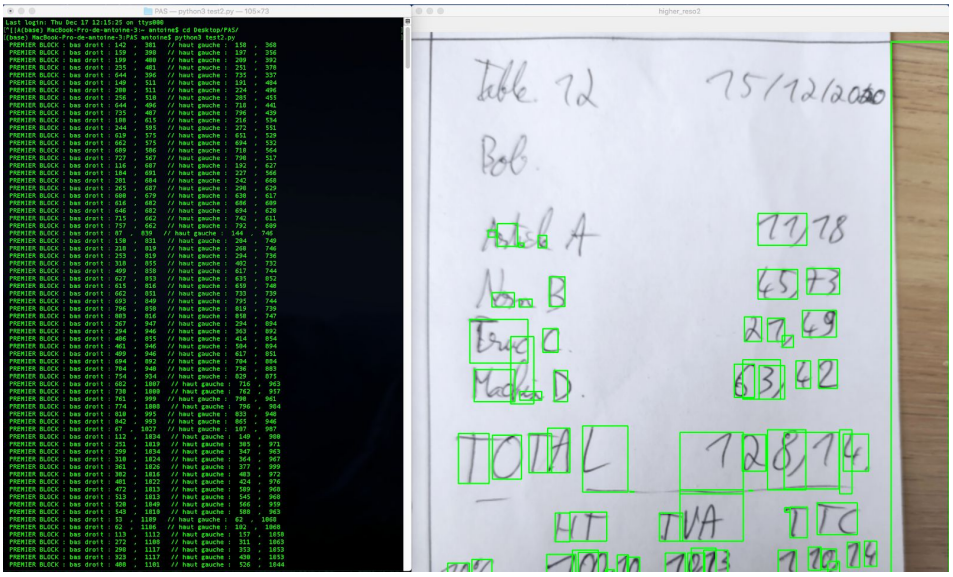
A la suite de ces traitement nous pouvons constater que le résultat de sortie de Tesseract est déjà mieux, en effet les deux premiers traitement effectués sur l'image permettent d'obtenir un meilleur retour, les deux derniers qui sont la fonction érosion et AdaptiveThreshold ne sont pas pertinente et de permettent pas d'améliorer la compréhension de l'image.

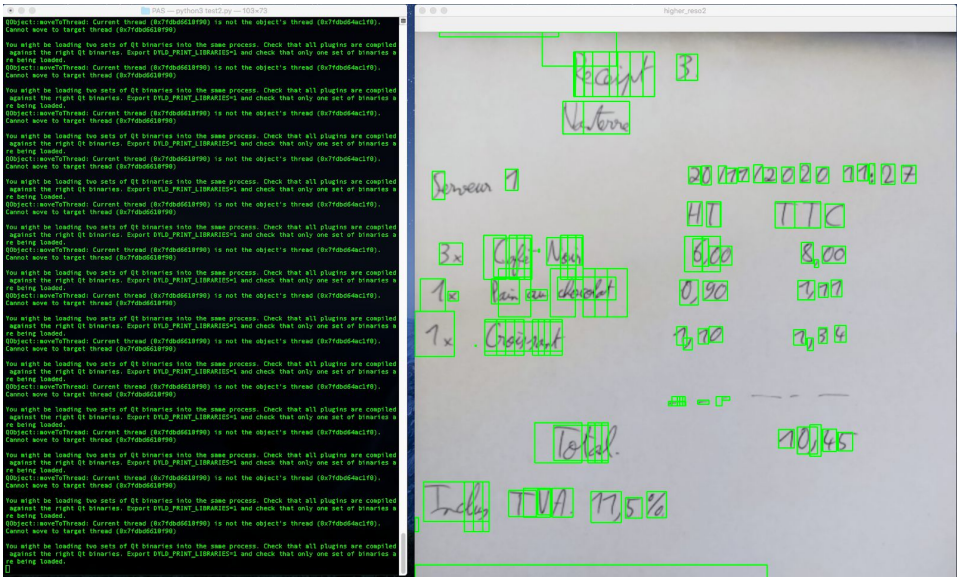
Le rendu n'étant donc pas pertinent avec les pré-traitement et la récupération direct du texte nous avons choisi d'essayer une seconde méthode et d'utiliser opencv2 qui est une librairie libre spécialisée dans le traitement d'image afin de détecter les zones de texte et à la suite d'en extraire le contenu. Cela va nous permettre de voir quels caractères sont détectés et ainsi comprendre les erreurs précédentes, en plus de cela nous allons afficher les coordonnées de chaque blocks. Cependant nous n'avons pu terminer ce script et finir de traiter les données par manque de temps.

Image avec affichage des boxes et des coordonnées




LE NID BAR
10, ALLEE DES GIROLLES
69120 MEUDON-LA-BEDOULE
LUNDI 20 MARS 2017
SERVI PAR : DAMIEN
SERVICE: AU BAR
PERRIER 3.50
CAFE EXPRESSO 2.00
DESTOP TURBO 8.00
DONT TVA 10%: 1.85 €
TOTAL TTC: 13.50 €
LA PROCHAINE FOIS,

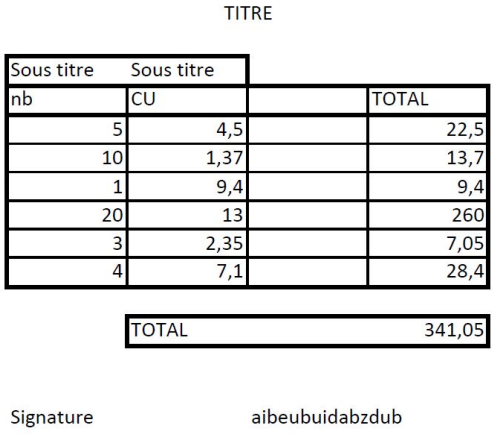




Travaux Google Vision

Par opposition à Tesseract qui offre de nombreux éléments de prétraitements, qu'il est nécessaire de maîtriser pour avoir des résultats pertinents, l'API Google Vision est une solution presque "Clef en main". En effet, le modèle est prêt à être utilisé d'un simple appel à l'API. Le résultat est un fichier JSON qui comporte de nombreux éléments, textuels, visuels...etc permettant de récupérer l'intégralité des informations. A partir de là, les problématiques rencontrées se trouvaient plus dans le tri des données récupérées et leur post-traitement pour avoir un résultat utilisable directement, par exemple en format CSV.

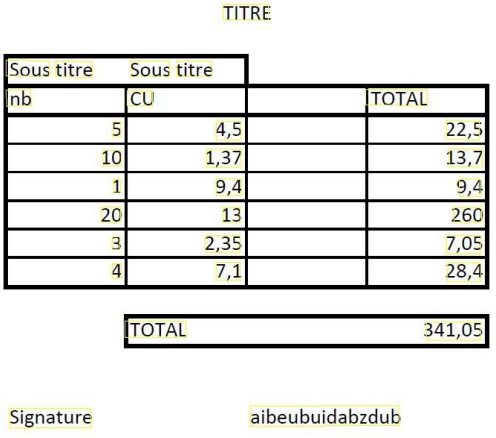
Dans un premier temps nous avons appliqué le traitement avec l'ensemble des informations sur les éléments les plus simples du jeu de données.



TITRE

Sous titre	Sous titre		
nb	CU		TOTAL
5	4,5		22,5
10	1,37		13,7
1	9,4		9,4
20	13		260
3	2,35		7,05
4	7,1		28,4
	TOTAL		341,05

Signature aibeubuidabzdub

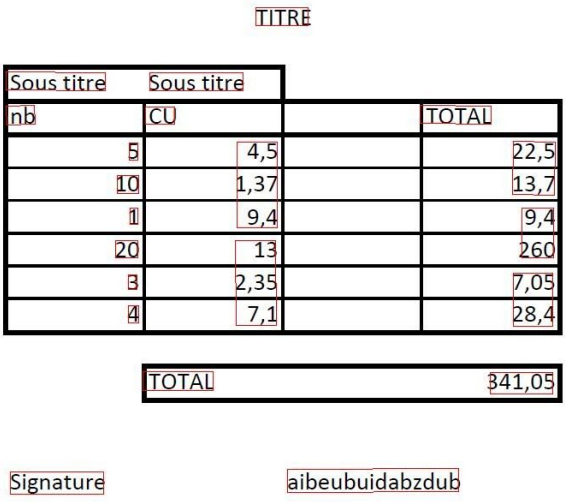


TITRE

Sous titre	Sous titre		
nb	CU		TOTAL
5	4,5		22,5
10	1,37		13,7
1	9,4		9,4
20	13		260
3	2,35		7,05
4	7,1		28,4
	TOTAL		341,05

Signature aibeubuidabzdub

A première vue, Google Vision détecte bien chaque élément, s'agissant d'un document "propre", cela est logique. Cependant, l'API fonctionne par paragraphe et block, qui encapsule les mots détectés, nous regardons donc à cette échelle les résultats sur les mêmes éléments.



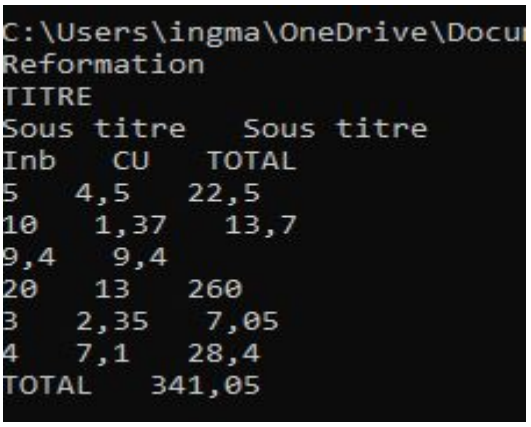
TITRE

Sous titre	Sous titre		
nb	CU		TOTAL
5	4,5		22,5
10	1,37		13,7
1	9,4		9,4
20	13		260
3	2,35		7,05
4	7,1		28,4
	TOTAL		341,05

Signature aibeubuidabzdub

Cette fois, on observe des regroupements assez incohérents, parfois d'une cellule, parfois de plusieurs, selon par exemple le nombre de caractères. A l'issue de cette première analyse, nous avons décidé de récupérer uniquement les mots et de reconstituer par nous même le tableau.

Ainsi chaque mot est défini par un bloc de coordonnées, ainsi que par son contenu sous forme de chaîne de caractères. Nous avons utilisé un algorithme de tri avec d'ordonner et de trier chaque bloc. Cependant si l'image est légèrement inclinée, deux éléments peuvent avoir des coordonnées légèrement différentes. De plus selon la résolution de l'image il était difficile de définir un écart précis à partir duquel il faut considérer qu'il s'agit d'une autre colonne ou une autre ligne. Ainsi nous avons fixé de façon arbitraire la limite à 5% de l'image totale. C'est à dire que si un élément est à plus de 5% de l'élément précédent sur l'axe horizontal, alors on considère qu'il s'agit d'une autre colonne, et de la même façon sur l'axe vertical pour les lignes.



TITRE

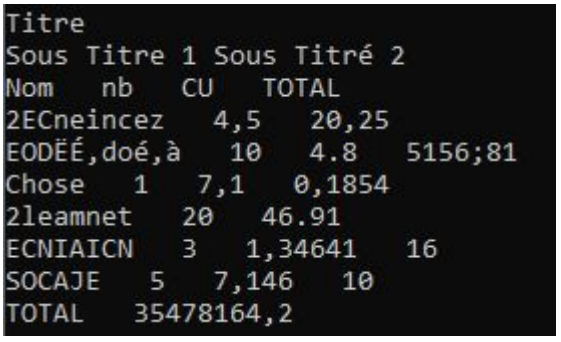
Sous titre	Sous titre		
nb	CU		TOTAL
5	4,5		22,5
10	1,37		13,7
1	9,4		9,4
20	13		260
3	2,35		7,05
4	7,1		28,4

TOTAL	341,05

Signature aibeubuidabzdub

On remarque quelques difficultés sur le traitement des bords noir et du chiffre 1.

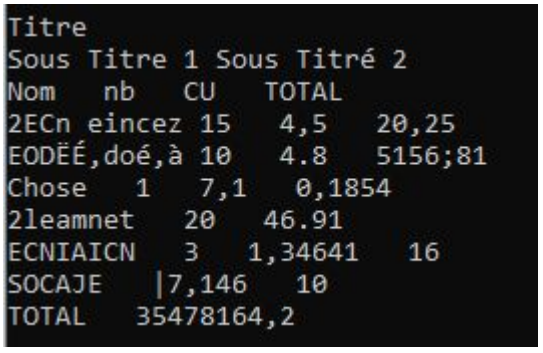
Nous avons appliqué le même traitement au jeu d'essai.



Titre

Sous Titre 1	Sous Titré 2		
Nom	nb	CU	TOTAL
2ECneincez	5	4,5	20,25
EODËé,doé,à	10	4.8	5156;81
Chose	1	7,1	0,1854
2leamnet	20	5	46.91
ECNIAICN	3	1,34641	16
SOCAJE	5	7,146	10
		TOTAL	35478164,2

SIGNATURE

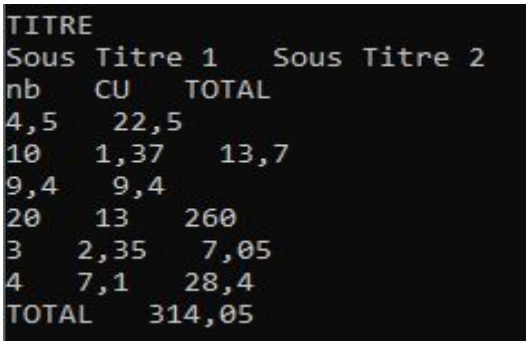


Titre

Sous Titre 1	Sous Titré 2			
Nom	nb	CU		TOTAL
2ECn eincez	5	4,5		20,25
EODËé,doé,à	10	4.8		5156;81
Chose	1	7,1		0,1854
2leamnet	20	5		46.91
ECNIAICN	3	1,34641		16
SOCAJE	5	7,146		10

TOTAL	35478164,2

SIGNATURE



Sous Titre 1	Sous Titre 2		
nb	CU		TOTAL
5	4,5		22,5
10	1,37		13,7
1	9,4		9,4
20	13		260
3	2,35		7,05
4	7,1		28,4

TOTAL 314,05

SIGNATURE

```
Grosse   Ber ghotel
38 18   Scheidegg
Familie Grindelwald   R. Müller
Rech.   Bar Nr. 4572   30.07.2007/13:29:17   Tisch   7/01
2xLatte   1xGloki Macchiato   à à   4.50   5.00 CHF   CHF   9.00
1xSchweinschnitzel à 22.00 CHF   5.00
1xChässpätzli   à 18.50 CHF   22.00   18.50
Total :   CHF   54.50
Incl. 7.6% MwSt   54.50 CHF:   3. 85
Entspricht   Es bediente in Euro   Sie: Ursula 36.33 EUR
MwSt   Tel.: Nr.:   033   430   853   234
Fax. : 033 853 67 67 16
```

Berghotel
Grosse Scheidegg
3818 Grindelwald
Familie R.Müller

Rech.Nr. 4572 30.07.2007/13:29:17
Bar Tisch 7/01

2xLatte Macchiato	à	4.50	CHF	9.00
1xGloki	à	5.00	CHF	5.00
1xSchweinschnitzel	à	22.00	CHF	22.00
1xChässpätzli	à	18.50	CHF	18.50

Total : CHF **54.50**

Incl. 7.6% MwSt 54.50 CHF: 3.85

Entspricht in Euro 36.33 EUR
Es bediente Sie: Ursula

MwSt Nr.: 430 234
Tel.: 033 853 67 16
Fax.: 033 853 67 19
E-mail: grossescheidegg@bluewin.ch

GHISONI, le Jeu 13 JUN 2019, 15:51

LE RESTO

Place De L'eglise 20227 GHISONI
Siret 53302164800019 TVA FR71533021648 NAF 56100
04 95 27 80 46

TK # 376 CAI1LV SIT53
VANNINA A votre service

OREZZA 1 L			4.50 (1)
BIERE PIETRA			4.50 (1)
BIERE HEINEKEN			
	2.80 x 2	:	5.60 (1)
CARRE D'AGNEAU			
	25.00 x 2	:	50.00 (1)
ENTRECOTE			18.00 (1)
COTES D'AGNEAU			17.50 (1)
MOUSSE A LA CHATAIGNE			
	6.50 x 2	:	13.00 (1)
TIRAMISU			7.50 (1)
TARTE AUX POMMES			6.50 (1)
CAVE ROUGE AOC 75 CL			14.00 (1)
CAFE EXPRESSO			
	2.00 x 4	:	8.00 (1)
GRAND CREME			
	3.00 - 100 % >		0.00 (1)

HT 135.54 Eur
TTC 149.10 Eur

TVA 1: 10.00% Tva:13.56 HT:135.54 TTC:149.10

12 Lignes

Reglement :MULTI R
Especes :50.00
Carte :99.10

LE RESTO vous remercie

KALIX - PAC-GEST V 4.17.2 B0173 - Sign : E E N K

```
GHISONI, le LE Jeu 13 RESTO   JUN 2019, 15:51
Siret   Place   53302164600019   De L'eglise   04 TUA   95   20227   FR71533021648   27 60 46 GHISONI WAF
TK # 376 CAI1LV SIT53   56100
VANNINA   OREZZA 1 L   A votre survice
BIERE   BIERE PIETRA   4.50 (1)
HEINEKEN   4.50 (1)
CARRE D'AGNEAU 2.80 x 2   5.60 (1)
ENTRECOTE   25.00 x 2   50.00   18.00 (1)
COTES D'AGNEAU   17.50 (1)
MOUSSE A LA CHATAIGNE   (1)
TIRAMISU   6.50 x 2   13.00 (1)
TARTE   CAVE AUX   ROUGE POMMES   7.50 (1)
CAFE EXPRESSO AOC 75 CL   6.50   14.00 (1)   (1)
GRAND CREME   2.00 x 4   :   8.00 (1)
3.00 - 100 X > 0.00 (1)
TTC   149.10   HT 135.54 Eur   Eur
TVA 1: 10.00% Tva:13.56 HT:135.54   TTC:149.10
Reglument   Especes   :HULTI R   12 Lignes
Carte   :50.00   :99.10
```

Tite de la Compogne Ffomatian
Ifomakion 20120/2020
N° Commande
12 opernue board

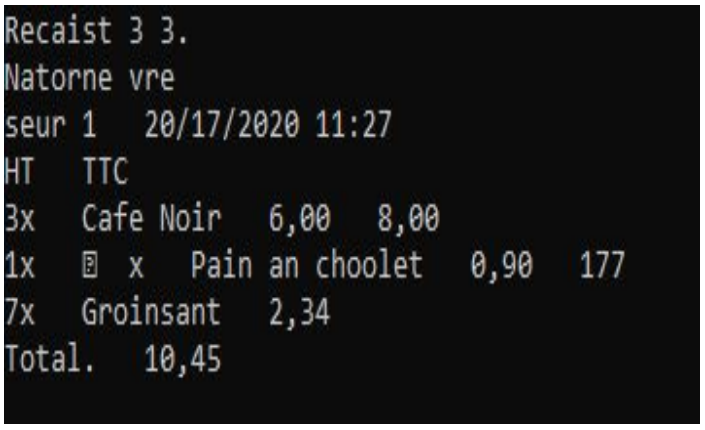
Code	label	Cout OI	TVAl	TTC
ZVA at	bltre	26,00	20	30,0
X 48	Bomicter	15,13	5	1720
ATG3	SW Mahie	66,12	12	7374

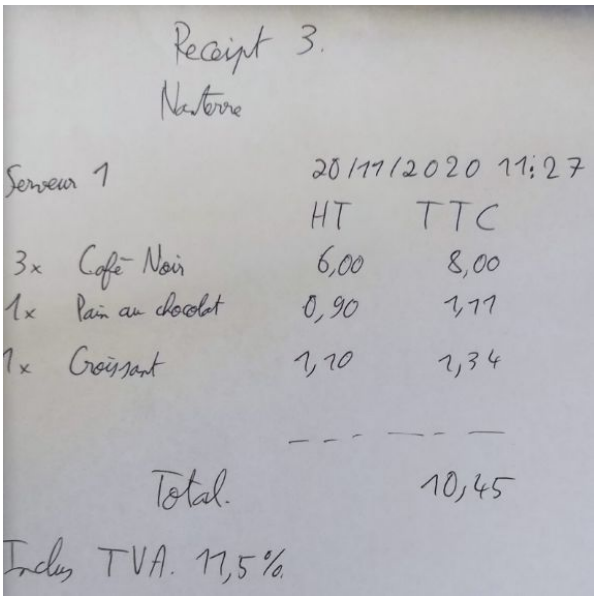
Titre de la Compagnie Information

Information 20/10/2020
N° Commande 12 avenue bérard

Code	label	Cout U	TVA %	TTC
ZVA	lote lettre	26,00	20	30,00
X48	Terminator	15,13	5	17,20
AT43	SW Machine	66,12	12	73,74

Total Inter € Total 125,37 €





Avec ces divers exemples, sur des tableaux excels, des tickets de caisse et des tableaux faits à la main, on remarque les limites de l'OCR et de notre traitement. D'une part la détection des éléments, notamment lorsque ces derniers sont manuscrits, malgré le réseau de neurones très performant de Google. Également sur la détection du chiffre 1 sur un tableau, il le considère parfois comme un séparateur. D'autre part dans le traitement des blocs on a parfois des écarts, avec la concaténation de 2 lignes sur une seule, la réduction des 5% décidé arbitrairement peuvent améliorer ce résultat mais le rendre trop précis peut inclure le passage d'une ligne à deux sans raisons.

Une fois les éléments extraits et triés avec plus ou moins de succès, la transformation en CSV est assez simple avec Python.

TITRE

Sous Titre 1	Sous Titre 2		
nb	CU		TOTAL
5	4,5		22,5
10	1,37		13,7
1	9,4		9,4
20	13		260
3	2,35		7,05
4	7,1		28,4
		TOTAL	314,05

SIGNATURE

TITRE
Sous Titre 1,Sous Titre 2
nb,CU,TOTAL
4,5,"22,5"
10,"1,37","13,7"
9,4,"9,4"
20,13,260
3,"2,35","7,05"
4,"7,1","28,4"
TOTAL,"314,05"

Conclusion

Ce sujet est vraiment vaste et compliqué à traiter sur un projet qui ne dure que 4 mois malgré une bonne organisation dès le début.
Cependant cela reste un sujet très intéressant et l'ensemble du travail réalisé a été très enrichissant et nous a permis dans un premier temps d'acquérir de nouvelles connaissances. En effet, nous n'avions pas de réels notions concernant les OCRs, de ce fait toutes les recherches nécessaires à l'étude de ce sujet nous ont permis de découvrir ce sujet et par la suite d'étoffer nos connaissances sur ce qui se fait dans ce domaine.
Concernant la partie mise en application et développement, c'est ici que nous avons manqué de temps car nous avons dû découvrir les outils à utiliser puis par la suite réaliser des programmes nous permettant de traiter les différentes données afin d'obtenir un résultat.

Nous pouvons également conclure sur les deux APIs utilisées, l'une étant une solution open-source (Tesseract) et l'autre payante pour une utilisation poussée (Google Vision) :

- Tesseract étant gratuit cela signifie qu'il y a beaucoup de travail à réaliser par l'utilisateur si ce n'est qu'il doit presque tout réaliser lui même, le traitement d'image, la réalisation des script, le formatage de sortie … Ce qui permet de comprendre le fonctionnement de A à Z cependant cela nécessite beaucoup plus de temps sur la prise en main ainsi que par la suite pour récupérer les informations des images.

- Google Vision étant quant à lui payant, la documentation est beaucoup plus simple et également plus poussée ce qui permet à l'utilisateur de gagner beaucoup de temps dans le traitement de ses images. Cependant bien qu'assez efficace, le côté "Boîte noire" de Google vision empêche toute modification ou compréhension des rejets. Ainsi il est nécessaire de favoriser le prétraitement pour appuyer la détection ainsi que le post-traitement afin de reconstituer les éléments lorsque le traitement les a perturbés.

Avec plus de temps nous aurions pu essayer l'API Vision Azure de Microsoft mais malheureusement un manque de temps nous en a empêché.

En conclusion, nous souhaitions comparer les différents outils actuels afin de s'appuyer dessus pour détecter et comprendre les tableaux le mieux possible. Ce sujet étant encore actuellement au cœur de nombreuses recherches, nos recherches ont mis en lumière les avantages de ces deux API Tesseract et Google Vision. Ainsi selon l'objectif attendu l'un ou l'autre est préférable : d'une part une API exigeante techniquement mais permettant une plus grande liberté, de l'autre une API boîte noire créée par l'empire Google, très entraînée et performante, mais dont les défauts sont plus difficilement corrigeables.
Enfin il est surtout important de se tourner vers les évolutions futures, qui amèneront probablement des réponses plus complètes à ce sujet. Par exemple, l'évolution de Google Vision en l'espace de 10 ans est impressionnante. Nous ne savons pas de quoi nous serons capables d'ici une autre décennie.