Étude comparative de reconnaissance de texte dans les bandes dessinées

Florian Le Meur*, Frédéric Rayar**,
Sylvie Treuillet***, Frédéric Daubignard****

* florian.lemeur@etu.univ-tours.fr, ** frederic.rayar@univ-tours.fr,
*** sylvie.treuillet@univ-orleans.fr, **** frederic.daubignard@algona.fr

Résumé. Cette étude se place dans le contexte de l'amélioration de l'accessibilité des livres, en particulier des bandes dessinées (BDs), aux publics empêchés de lire. A cette fin, la reconnaissance automatique de texte dans les BDs est une tâche fondamentale dans l'analyse de ces dernières. Nous proposons ici une étude comparative de différents algorithmes de segmentation et de reconnaissance de texte dans des BDs à partir d'images acquises à main levée à l'aide d'un terminal mobile. Nos expériences, réalisées sur une base de données créée spécifiquement pour cet usage, ont permis de retenir les méthodes les plus prometteuses, et de les intégrer au sein d'une application mobile, destinée aux personnes souffrantes de troubles de la lecture.

1 Introduction

La lecture est un outil primordial dans le développement de la connaissance et du savoir, mais aussi une activité ludique pour l'épanouissement des enfants, des adolescents, ou des adultes. Le marché des bandes dessinées (BD), en particulier, connaît un développement très important ces dernières années. Cependant cette activité enrichissante n'est pas accessible à un certain nombre de personnes, entrant dans la catégorie dite des "publics empêchés de lire" (dyslexie, autisme, déficience visuelles,...). Dans cette étude, nous cherchons à améliorer l'accessibilité des bandes dessinées en proposant une lecture augmentée à partir d'un terminal mobile tel qu'un smartphone ou une tablette. Tout en maintenant le contact physique avec l'objet livre, cette application mobile d'aide à la lecture répondrait aux contraintes suivantes : (i) travailler à partir de flux vidéos en provenance d'un terminal mobile (cela implique des difficultés liées à la luminosité, à l'angle de prise de vue de la caméra) et (ii) fournir une interface la plus ergonomique possible permettant un paramétrage simple pour créer des profils utilisateurs. Dans cet article, nous présentons un état de l'art et une étude comparative pour la reconnaissance de texte dans des bandes dessinées à partir d'images acquises à main levée à l'aide d'un terminal mobile. Une chaîne de traitement est proposée, de façon modulaire, de manière à pouvoir évaluer unitairement les algorithmes retenus Étude comparative de reconnaissance de texte dans les bandes dessinées et sélectionner les meilleurs dans notre cadre applicatif. En particulier, nous nous concentrons ici sur les modules de segmentation de texte (détection) et de reconnaissance de texte. Afin de confronter ces algorithmes, une première base d'images de

BD hétérogène et non complaisante, issues d'ouvrages variés en français et en anglais, a été créée. La suite de l'article présente un état de l'art des méthodes de segmentation de texte et de reconnaissance de texte dans les bandes dessinées en section 2. La chaîne de traitement de notre application d'aide à la lecture et la base de données créée sont ensuite détaillées dans la section 3. Puis,les résultats des expériences conduites sont présentés et discutés dans la section 4. Enfin, la section 5 résume les contributions apportées et répertorie les perspectives de ces travaux.

2 État de l'art

2.1 Segmentation dans les bandes dessinées

Différentes approches existent pour segmenter et isoler du texte de bandes dessinées. La première approche consiste à utiliser une technique générique de segmentation de texte destinée à l'usage dans des scènes naturelles et à l'appliquer sur des BD. Certaines techniques modernes telles que CRAFT développée par Baek et al. (2019) atteignent ainsi des performances supérieures à 90% de score F1 en utilisant des réseaux de neurones convolutifs spécialisés permettant de segmenter une image (VGG et Unet dans le cas de CRAFT). Cependant, l'application de techniques génériques destinées à du texte de scènes naturelles sur des BD entraîne une réduction des performances comme a pu le montrer l'étude de Rayar et Uchida (2019). L'approche développée récemment par Del Gobbo et Matuk Herrera (2020) utilise également un réseau de neurone convolutif (basé sur ResNet et Unet) entraîné pour détecter le texte au pixel près dans des mangas. La performance de cette solution spécialisée pour les mangas avoisine les 90% de score F1 mais peut être sensiblement inférieure sur d'autres types de BD. Une autre approche utilise également des réseaux de neurones convolutifs mais cette fois pour segmenter le phylactère (la bulle) autour du texte et non le texte directement. Cette technique spécialisée pour l'application sur des BD est développée par Dubray et Laubrock (2019) et atteint également 90% de score F1. L'inconvénient de cette technique est d'être dépendante des bulles et de leur forme dans l'ouvrage et n'est pas appropriée pour détecter le texte en dehors des bulles et certaines onomatopées.

2.2 Reconnaissance de texte dans les bandes dessinées

Il existe un certain nombre de techniques permettant de reconnaître du texte générique dans des images qui sont applicables sur des bandes dessinées. En plus de réussir à lire du texte typographié, ces programmes appelés OCR (*Optical Character Recognition*) sont également capables depuis quelques années de reconnaître du texte manuscrit grâce à l'utilisation de réseaux de neurones. Tesseract [1] par exemple est un OCR développé depuis plusieurs dizaines d'années qui intègre depuis 2019 des réseaux de neurones convolutifs et récurrents pour augmenter ses performances, notamment sur les textes manuscrits. Un autre OCR, Calamari, développé depuis 2018 par Wick et al. (2020) utilise des réseaux de neurones spécifiquement entraînés pour la reconnaissance de textes manuscrits anciens (XIII^{e} - XIX^{e} siècles). Enfin, il

1. https ://github.com/tesseract-ocr

existe d'autres techniques développées par des sociétés comme ML-Kit [2] et Shape Detection [3] qui sont des solutions développées par Google. L'inconvénient de ces techniques est qu'elles sont souvent peu documentées et le code source rarement disponible.

3 Méthode proposée

3.1 Base de données

Afin de pouvoir évaluer les performances des différentes techniques en conditions réelles nous avons établi une base de 50 photographies de bandes dessinées issues de 14 ouvrages différents : BD franco-belges, mangas, comics et un roman graphique (graphic novel). La répartition entre les différents types de BD est indiquée sur la Figure 1. Les photos ont une résolution de 480x640 pixels et sont réalisées par un smartphone tenu horizontalement à environ 6 cm au-dessus de la vignette ciblée. L'appareil photo a une longueur focale de 3.81 mm.

FIG. 1: Part de chaque collection d'ouvrages dans la base

FIG. 2: Exemples de bulles classiques et spéciales

Chaque photographie contient une bulle ou éventuellement 2 bulles liées ou superposées. Nous avons séparé les bulles en 2 catégories : les bulles classiques et les bulles complexes. Les bulles classiques sont plus couramment rencontrées. Elles contiennent entre 50 et 150 caractères soit une vingtaine de mots en moyenne. Elles n'ont pas de marques graphiques spécifiques et la police utilisée est facilement lisible. Chaque ouvrage de la base a au moins une bulle classique. Les bulles complexes sont des bulles moins courantes et plus difficiles à reconnaître pour un algorithme. Les contours des bulles peuvent être en pointe, le fond de la bulle coloré, le texte cisaillé ou déformé. Les bulles classiques ne représentent que 24% de la base de test et sont donc largement sous-représentées par rapport à un ouvrage complet. Pour

2. https ://developers.google.com/ml-kit/
3. https ://wicg.github.io/shape-detection-api/

cette raison, les résultats obtenus en travaillant sur cette base sont plutôt des résultats minorés par rapport à ce que l'on obtiendrait sur une bande dessinée complète.

3.2 Chaîne de traitements

La transcription audio du texte à partir d'une image se décompose en 4 étapes présentées sur la Figure 3. La segmentation pour isoler le texte, la reconnaissance des caractères (OCR), une étape de correction et enfin la transcription audio du texte avec un synthétiseur vocal. Notre étude comparative s'est principalement focalisée sur les étapes de segmentation (algorithme Del Gobbo) et d'OCR (ML-Kit, Tesseract, Calamari et Shape Detection), les choix de ces algorithmes étant justifiés par des tests préliminaires que nous ne pouvons reporter dans le présent article, faute de place. L'apport de l'étape de correction semble faible mais l'étude sur les correcteurs n'a pas été approfondie. La dernière étape de Text-to-Speech ne rentre pas dans le cadre de notre étude. Le moteur de Text-to-Speech utilisé est celui de Google, intégré à Android.

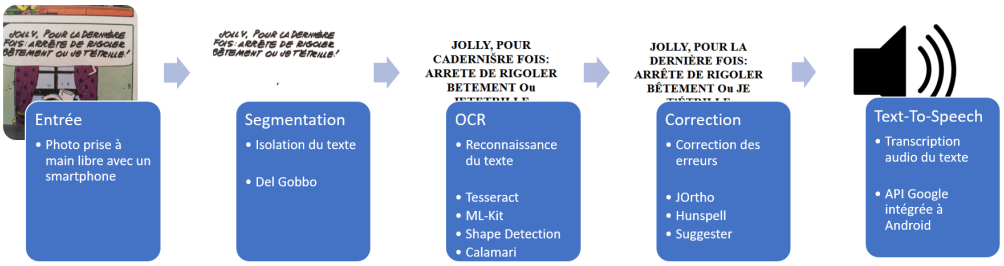

FIG. 3: Pipeline utilisé pour l'étude comparative

4 Évaluation

4.1 Métriques utilisées

Les métriques utilisées pour comparer 2 chaînes de texte et mesurer leurs différences sont classiquement le taux de caractères erronés (Character Error Rate / CER) et le taux de mots erronés (Word Error Rate / WER). Ces métriques se basent sur la distance d'édition ($dist$) entre 2 séquences de caractères (s_{ref}, s_{rec}) normalisée par la taille de la séquence de référence (s_{ref}).

$$CER, WER = \frac{dist(s_{ref}, s_{rec})}{taille(s_{ref})}$$

La distance d'édition est calculée différemment : pour le CER les erreurs sont mesurées à l'échelle d'un caractère tandis que pour le WER elles le sont à l'échelle d'un mot. Dans les 2 cas, la valeur obtenue est 0 si les 2 séquences sont identiques ; 1 si toutes les lettres ou tous les mots sont erronés et les taux peuvent être supérieurs à 1 en cas de nombreux faux-positifs dans la séquence reconnue (s_{rec}).

Type de test		CER moyen	WER moyen
Segmentation	Photo	0.30 ± 0.23	0.58 ± 0.33
	Segmentation	0.34 ± 0.26	0.63 ± 0.28
OCR sur images segmentées	Shape Detection	0.40 ± 0.19	0.63 ± 0.28
	Tesseract 4	0.34 ± 0.26	0.63 ± 0.28
	ML-Kit	0.20 ± 0.12	0.52 ± 0.26
OCR sur images brutes	Tesseract 4	0.30 ± 0.23	0.58 ± 0.33
	ML-Kit	0.16 ± 0.13	0.48 ± 0.30

TAB. 1: Résultats mesurés

4.2 Résultats

Apport de la segmentation : L'intérêt d'une étape préalable de segmentation est évaluée indirectement par les performances en sortie d'un OCR, ici Tesseract 4. La première partie de la Table 1 compare les résultats obtenus par l'OCR appliqué directement sur la photo brute en entrée ou après segmentation de celle-ci. Nous observons que la segmentation apporte peu d'améliorations et les CER moyens sont très proches, 0.30 contre 0.34. L'analyse des résultats collection par collection sur la Figure 4 montre que l'impact de la segmentation est assez variable. Dans certaines collections comme les 3 premières, *Tintin*, *Boule & Bill* et *Les Schtroumpfs* l'impact est quasi-inexistant. Dans *Lucky Luke* ou *Les Mystérieux Mystères Insolubles* l'impact est très négatif tandis que dans *Astérix* et *The Walking Dead* l'impact est plutôt positif. Il ne semble pas y avoir de critère évident permettant de déterminer ce qui influence positivement ou négativement les résultats de la segmentation. Globalement sur les bulles classiques (bulles les plus rencontrées dans les BD) l'étape de segmentation améliore la reconnaissance.

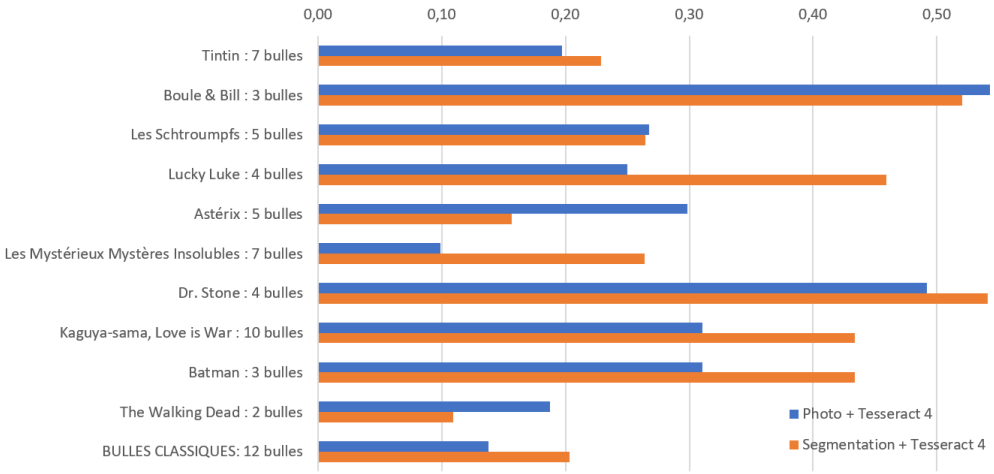

FIG. 4: Résultats (CER) obtenus avec et sans segmentation pour chaque collection

(a) Image brute (b) Image segmentée

FIG. 5: Exemple de segmentation nuisant à la reconnaissance

Dans l'exemple présenté sur la Figure 5 on remarque que la segmentation présente des erreurs. L'un des mots n'est pas segmenté (faux négatif) et le fond de la bulle n'est que partiellement segmenté par endroits. Quelques autres éléments sont des faux positifs qui ne devraient pas apparaître sur l'image segmentée. Pour ces raisons, de plus mauvais résultats sont obtenus sur l'image segmentée que sur l'originale (0.4 contre 0.05 [4]).

Comparaison des OCR sur images segmentées : La seconde partie de la Table 1 présente les résultats des 3 OCR testés (Shape Detection, Tesseract 4 et ML-Kit) sur les images segmentées en amont par l'algorithme de Del Gobbo. Les performances de Calamari n'ont pas été présentées ici car son utilisation nécessite au préalable une séparation des lignes de texte, ce qui n'est pas simple à mettre en place sur un support comme une BD où le texte est souvent irrégulier. Des essais préliminaires ont donné des valeurs de CER élevées (> 0.5). La Figure 6 compare les performances collection par collection. Il apparaît que ML-Kit obtient un CER plus bas que Tesseract et Shape Detection sur presque toutes les collections. Cette différence est plus marquée sur les 5 dernières listées en Figure 6. Comme les 5 premières (De *Tintin* à *Astérix*) présentent une écriture manuscrite tandis que dans les 5 dernières (Des *Mystérieux Mystères Insolubles* à *The Walking Dead*) une écriture typographiée, il apparaît donc que ML-Kit aurait une meilleure performance sur les textes typographiés que Tesseract qui obtient des résultats plus homogènes sur l'ensemble de la base. Cette différence de performances entre textes manuscrits et typographiés est encore plus marquée avec Shape Detection qui obtient des CER très élevés sur les BD manuscrites. Dans l'ensemble, les bulles classiques présentent des valeurs plus faibles que les autres collections ce qui signifie que ces résultats sont plutôt minorés par rapport à ce que l'on obtiendrait sur une bande dessinée complète.

Comparaison des OCR sur images brutes : Sur des photos brutes (non segmentées), Shape Detection n'a pas été capable d'identifier les caractères. On constate sur la Figure 7 que Tesseract et ML-Kit obtiennent des résultats similaires sur les collections manuscrites à l'exception de *Boule & Bill* qui ne représente que 3 bulles. Sur les BD typographiées on retrouve les différences observées lors des essais sur les images segmentées où ML-Kit s'est avéré plus performant. La performance moyenne est sensiblement meilleure que lors des essais sur les images segmentées. Pour Tesseract, le CER est de 0.30 contre 0.34 avec segmentation. Pour ML-Kit,

4. A noter que le texte considéré est uniquement celui de la bulle en rouge et que sa taille réduite amplifie le CER/WER

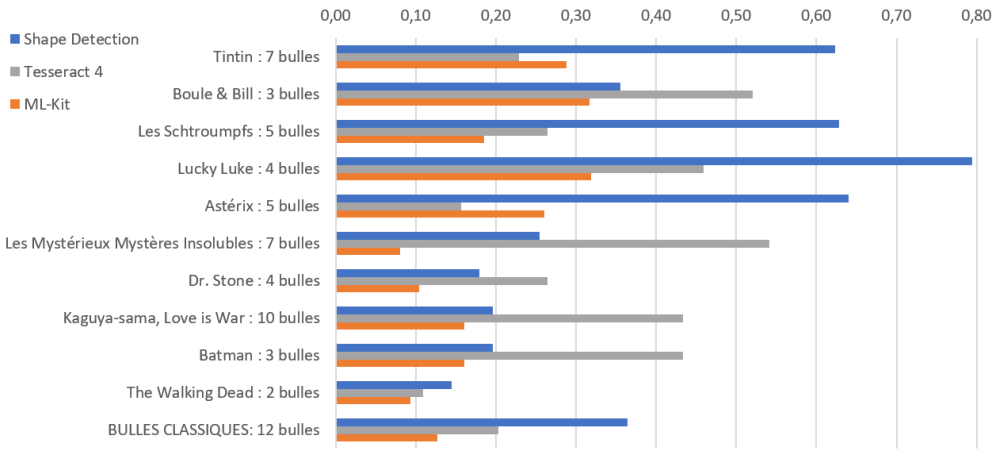

FIG. 6: Résultats (CER) obtenus avec les différents OCR sur des images segmentées

il est de 0.16 contre 0.20 avec segmentation. La segmentation aurait donc, en moyenne, un impact sensiblement négatif mais celui-ci est très variable selon les collections. Sur les essais, ML-Kit réalise globalement de meilleures performances que Tesseract en obtenant des taux d'erreurs moyens plus faibles. On constate que pour les images ayant un CER ≤ 0.05 les erreurs sont souvent mineures et concernent le plus souvent des symboles comme les " !" et des points soit manquants soit ajoutés. Ces erreurs sont d'une faible gravité et n'influencent pas toujours le texte prononcé lors de l'étape de Text-To-Speech. Pour les images ayant un CER entre 0.05 et 0.15 les erreurs peuvent être plus ou moins gênantes pour la lecture selon leur position dans le texte. Enfin, pour les images ayant un CER ≥ 0.15 le texte reconnu est souvent significativement dégradé et il est souvent difficile d'en comprendre une partie.

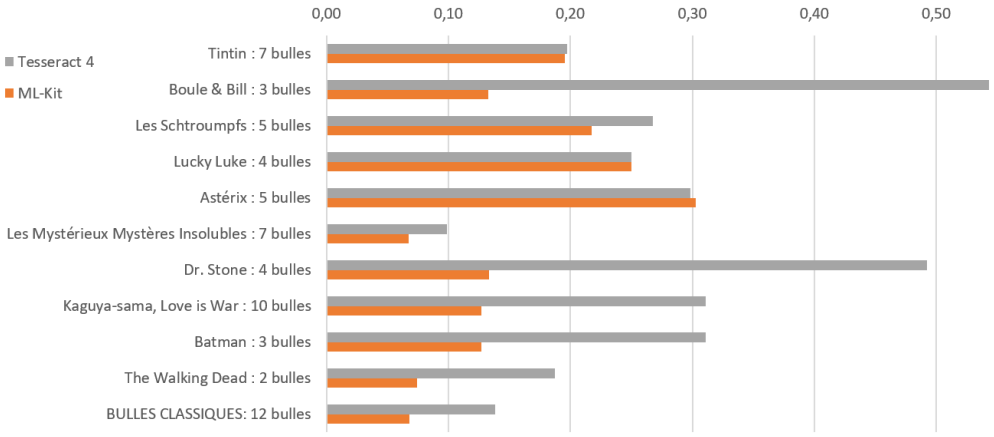

FIG. 7: Résultats (CER) obtenus avec les différents OCR sur des images brutes

5 Conclusion et perspectives

Nous avons présenté dans cet article une étude comparative pour la reconnaissance de texte dans des bandes dessinées. L'objectif de cette étude est de développer une application mobile pour améliorer l'accessibilité aux livres des personnes en situation de handicap. Dans cet article, nous avons donc (i) proposé une chaîne de traitement d'aide à la lecture, (ii) implémenté une application *proof-of-concept* mettant en œuvre ladite chaîne de traitement et (iii) construit une base de données pour évaluer les meilleures méthodes de l'état de l'art. Les premiers résultats obtenus sont intéressants mais il apparaît évident que la qualité actuelle de la reconnaissance de texte ne permet pas une utilisation dans l'état. Nous envisageons de creuser deux pistes d'améliorations dans la suite de nos travaux : l'utilisation d'une architecture *"end-to-end"* qui réaliserait la segmentation et la reconnaissance de texte en une étape, et l'étude plus poussée des possibilités des correcteurs orthographiques pour améliorer la qualités des transcriptions obtenues.

Références

Baek, Y., B. Lee, D. Han, S. Yun, et H. Lee (2019). Character region awareness for text detection. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 9357–9366. IEEE Computer Society.

Del Gobbo, J. et R. Matuk Herrera (2020). Unconstrained text detection in manga : A new dataset and baseline. In *Computer Vision – ECCV 2020 Workshops*, pp. 629–646. Springer International Publishing.

Dubray, D. et J. Laubrock (2019). Deep cnn-based speech balloon detection and segmentation for comic books. In *2019 International Conference on Document Analysis and Recognition (ICDAR)*, pp. 1237–1243. IEEE Computer Society.

Rayar, F. et S. Uchida (2019). Comic text detection using neural network approach. In *Multi-Media Modeling*, pp. 672–683. Springer International Publishing.

Wick, C., C. Reul, et F. Puppe (2020). Calamari - A High-Performance Tensorflow-based Deep Learning Package for Optical Character Recognition. *Digital Humanities Quarterly 14*(1).

Summary

This paper addresses the improvement of books' accessibility, especially comics, for people that have difficulty to read. To do so, we propose a comparative study of several text segmentation and text recognition in comics from images acquired freehand using a mobile terminal. Our experimentations, performed on a database that has been curated for our purpose, have allowed us to isolate the most relevant methods, use them in a modular workflow, and implement and proof-of-concept mobile application.