A NOVEL APPLICATION OF OPTICAL CHARACTER RECOGNITION FOR PRODUCT IMAGE COMPLIANCE

Maharshi Dutta
Abhinav Chanda
Samir Husain
Tirthankar Mukhopadhyay
Matthew A. Lanham

Contents

A Novel Application of Optical Character Recognition for Product Image Compliance

Maharshi Dutta, Abhinav Chanda, Samir Husain, Tirthankar Mukhopadhyay, Matthew A. Lanham
Purdue University, Department of Management, 403 W. State Street, West Lafayette, IN 47907
dutta33@purdue.edu; chanda0@purdue.edu; husain2@purdue.edu; tmukhopa@purdue.edu;
lanhamm@purdue.edu

ABSTRACT

Any retail company with a digital platform has multiple legal/internal compliances that need to be verified for the images that are displayed on the website. This work is an attempt at creating a solution to automate the audit process. In addition to protecting the loss due to potential lawsuits in case of non-compliant listings, the solution aims at improving resource utilization by 50% for a major US retail company spent on the manual auditing process required for ADA compliance check. Moreover, it helps prevent lost business opportunities because of returns forced due to incorrect listings – leading to significant cost benefits up to millions of dollars. The entire project is implemented using Python with an object-oriented approach. The BeautifulSoup package has been used to scrape the images from the digital platform. Optical character recognition has been implemented using Pytesseract to extract label texts from images. A business rule validation framework was created to provide a generic solution that is scalable. Thus, this solution can be extended to any industry facing a similar challenge.

Keywords: compliance, auditing, automate, object-oriented, Python, BeautifulSoup, scrape, Pytesseract, optical character recognition, generic

1. INTRODUCTION

Optical Character Recognition (OCR) is a computer vision challenge. Work that was done in the field of OCR dates to 1914, much earlier than the data analytics and consequent artificial intelligence boom. The meteoric rise of AI, enabled by the tremendous computational power, has breathed new life into the field of OCR. Deep learning techniques can be used to perform OCR "in the wild" with very high levels of accuracy.

Our client – a major US retailer – has an e-commerce platform with an exhaustive list of products. The organization sells household products, food, and drugs. Each product has an image(s) on the digital platform. Under US/internal compliance requirements, product images on a digital platform need to show clearly,

- All warnings - such as hazardous, choking, flammable
- Nutritional information in the case of edible products
- Chemicals, drug strength and side effects in the case of drugs

We devised an intelligent, automated system to read product listings from a retailer's website, analyze if they met compliance standards, extract required information, and flag non-compliant listings for escalation. Our algorithm can identify both sparse and dense texts, different font types, and non-standardized text structures.

In the current process, the organization uses a custom OCR approach to classify the necessary information available on the product image. As per this process, they source the picture of each product from the supplier

and use an existing OCR software to convert the image into digitized text. Once the photos have been processed into segmented texts, the image compliance check and updating of flags are done manually.

The current process leads to high costs for the organization. They want an automated solution that -

1. Checks whether an image is compliant or not based on warning labels and internal business rules such as size, angle, clarity
2. Notifies suppliers if the pictures are non-compliant
3. Reduces manual co-ordination with the supplier post-compliance check
4. Saves time spent in correcting the listings. Longer times lead to delay in the product listing and a potential loss of business

The solution automates this whole process to reduce manual intervention and efforts being spent on the above operation.

A study suggests that by 2024, 70% of the consumers will buy groceries online, leading to a $100Bn market (MiBiZ,2018). In this digital age, automation is vital for any player to be relevant in any market. The methodology that has been adopted focuses on creating a generalized solution. Using OCR as the backbone, our algorithm is extendable to handle any requirement where information is needed to be captured from the product images on a website. With an initial focus on mandatory product information that is required by compliance laws, the solution is capable of being extended to extract product information like brands and product descriptions.

The solution can also be used by various government agencies to check for compliance concerning the mandatory information that needs to be printed on the product package. As numbers suggest, with growth in the digital universe, automation not only makes it feasible for agencies to keep track of the ever-growing online marketplace but also improves accuracy leading to smooth governance.

As part of our solution, we automate the entire process. The algorithm can,

1. Scrape data from a supplier website sourcing the required product images
2. Apply a deep learning algorithm to check whether these images are compliant with our client's needs
3. If images are non-compliant, the suppliers are notified with a list of compliance checks
4. Out of the non-compliant images, our algorithm accurately identifies risk listings, nutritional info, and drug fact

We worked on providing two key innovations which would help the client as well as any organization in the industry required to comply with these rules,

- The primary feature of our algorithm is its ability to factor in all the regulations together and classify a product as compliant or not. This would help our client practically eliminate any manual effort being spent post the OCR software processing
- Another feature of the algorithm is its generic nature. Any set of compliance rules, when fed into the process, would be factored in and the machine learning algorithm would predict the compliance based on these rules

This is a crucial feature, as it lends scalability to our solution, such that it be used and deployed by any retailer in any market with minimal code changes.

The applied machine learning solution would help the organization save costs by cutting down on approximately 6000-man hours of audit in addition to protecting the loss due to potential lawsuits in case of non-compliant listings. It also helps prevent lost business opportunities because of returns forced due to incorrect listings – both can lead to significant cost benefits up to millions of dollars.

2. LITERATURE REVIEW

The unique challenge that has been solved involves three primary processes:

1. Web-scraping to source images.
2. Text Extraction
3. Compliance Check

An extensive literature survey on a wide spectrum of work from a timeline ranging from the late 1990s has been considered to develop pivotal ideas of the solution.

2.1 Web Scraping

Several web scraping techniques like HTTP programming, HTML parsing, DOM parsing, a pre-built tool like Mozenda, etc. have been discussed in detail in a paper titled 'An Overview On Web Scraping Techniques And Tools' [1]. HTML parsing had the best balance between functionality and ease of use out of all the suggested techniques. Hence, web scraping was implemented using this technique.

2.2 Text Extraction

An extensive study on literature available for text extraction from images was done to identify the challenges involved in the process.

2.2.1 Image Pre-processing

Better the image, better is the OCR performance. This has been evidenced in the work titled 'Image Pre-processing for Improving OCR Accuracy' [2]. The technique of contour detection was improved to detect spots in a microscopic image. Resolution normalization, noise reduction using low pass filters and reconstruction of boundaries helped in detecting circular contours improved canny method of contour detection. This has been considered in the design for implementing an object detection approach to crop relevant portions of an image. In data exploration, multiple skewed images were observed. Work carried out in 1995 titled 'Measuring document image skew and orientation' introduced the idea of measuring image skewness comparing characters from a dictionary with detected characters to calculate skewness by considering the orientation of the standardized character [3]. Another approach to handle image skewness is the use of 2D Haar Wavelet Transformations [4][5]. This approach was factored into the image pre-processing stage. Instead of using standardized characters, edge detection was used to determine the skewness of an image. Contrast adjustment and thresholding were techniques used by a group of academics to pre-process images [6]. This yielded an 80% accuracy rate of detecting labels on images.

2.2.2 Optical Character Recognition

In 2012, work was conducted to compare the performance of various algorithms in feature extraction for character recognition [7]. A blend of multiple probabilistic neural networks yielded an accuracy of 94.12%. This planted the idea of using a blended model instead of a standalone model. The use of a convolutional neural network was also considered based on a 2015 paper titled 'Character-level Convolutional Networks for Text Classification' [8]. In CNN, hyperparameter tuning of epochs and learning rate improves model performance [9]. While exploring various object detection approaches, the use of median filters to detect nutrition labels seemed to be an efficient technique leading to a very high accuracy rate [10]. Another aspect that was observed during data exploration was the presence of multiple images for a single product. Another interesting work was done in 2014 to accumulate predictions made from multiple images of the same object [11]. A similar approach has been taken in this work as well to accumulate OCR outputs for a single product.

2.3 Compliance Check using OOP

Scalability is an important aspect of any solution. The solution was conceptualized as a modularized pipeline with well-defined dependency and exception handling. The fundamentals of an object-oriented approach, as discussed in a 2018 paper titled 'Object-Oriented Programming in Computer Science', were used in the architecture design [12]. To implement efficient search operations, hash tables were proposed in a paper titled, 'A study on the usage of data structures in information retrieval' [13]. Exception handling was incorporated keeping in mind the principles laid down in a paper titled 'Exception Handling: A Field Study in Java and .NET' [14].

Section	Topic	Previous Work Done	Novel Application
2.1	Web Scraping	HTML, HTTP, DOM parsing- different techniques to scrape unstructured data from web	Custom Scraping library built to scrape images from a website into a structured format.
2.2.1	Image Pre-processing	Measure Image skewness using 2D Haar Wavelet Transformations.	Used geometrical transformation by interpolation methods.
		Binary thresholding or its variants have been used in standalone models.	A blend of binary and OTSU thresholding instead of using a single model.
		Contrast Adjustment	Custom function written to identify threshold point which will lead to maximum contrast.
2.2.2	OCR	Blend of probabilistic neural network used.	An optimized LSTM model was used.
		CNN used for object detection.	Connectionist Text Proposal Network, variation of a fast RNN was used.

Figure 2.1 Literature Review Summary

3. DATA

The model has been run on images of products belonging to seven product categories. Images of 14,000 products were scrapped from the digital platform of the company, each product has multiple images. In figure 3.1, the proportion of products in each product category that has been used to tune the pipeline has been given. A stratified sampling approach has been used to maintain the proportion of data is in accordance with the data that is available on the digital platform.

Product Categories	% of Products
FRESH	8.44%
GROCERY	25.80%
HARDLINES	17.06%
HEALTH AND BEAUTY	24.28%
IN AND OUTDOOR HOME	13.27%
PETS AND CONSUMABLES	6.41%
SOFTLINES	4.73%

Figure 3.1 Proportion of products used from each category

Column Name	Column Description
Product ID	Uniquely identifies a product
Product Name	Name of the product
Level 6 Category	Highest level in product hierarchy
Level 5 Category	Product Hierarchy level
Level 4 Category	Product Hierarchy level
Level 3 Category	Product Hierarchy level
Level 2 Category	Second last level in product hierarchy

Figure 3.2 Metadata

4. METHODOLOGY

An end to end solution to check ADA compliance with scalability and usability has been the central principle around which the methodology has been designed. Figure 4.1 shows a schematic representation of the entire workflow.

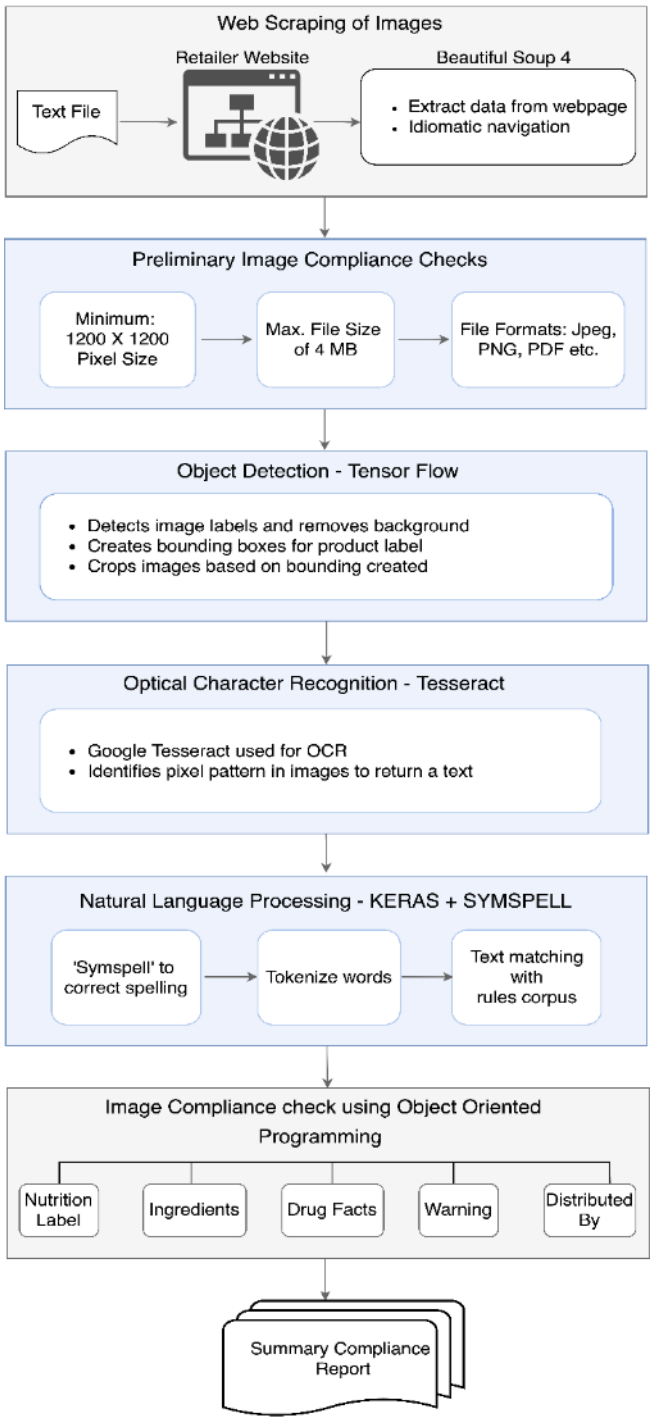

Figure 4.1 Scheme diagram for end to end solution

4.1 Web Scraping

A csv file has been used to get the product codes for which images have been extracted from the website. An object identification approach with relevant tag checks has been used to download the images of a product. Beautiful Soup, a Python package for parsing HTML and XML documents, has been used for the purpose. Scrapy package had been initially considered. However, the user-friendly framework provided by Beautiful Soup enabled to design a structured approach in data extraction. Relevant web page status checks have been made to ensure proper exception handling. Product listings with no actual digital presence or with no front-facing image have been reported in a CSV file. The products with at least one front-facing image have been downloaded in a folder with the product code as the folder name. This ensured compatibility with the next steps in the pipeline.

4.2 Object Detection

A custom library was created to detect relevant portions of the image which would be fed into the optical character recognition engine. There were two classes of cropping that was included in this library. The first one was to specifically locate certain labels. This has been implemented using tensor flow. Graphs have been used to detect boxes. These boxes were scored according to relevance. Using cutoff values, coordinates of these boxes have been located, which were used to crop the desired portion of the image. The second class of cropping has been built using the OpenCV module. The edges of an image were detected and used to crop. This was done to enhance the performance of the OCR engine.

4.3 Optical Character Recognition

A custom library was created for optical character recognition. The class included functions for handling skewness, image thresholding, greyscale conversion, and text processing. OpenCV has been used to handle skewness, using geometrical transformations. For image thresholding, a blended approach has been adopted. First, the threshold providing for the highest contrast is identified, followed by binary and OTSU thresholding, which gave better results than standalone models. Image pre-processing was given high importance because the quality of text extraction depends directly on the quality of the image. The pre-processed image is then fed into Pytesseract. The text output is then processed using SymSpell. The text is tokenized into two-word sequences. Then, utilizing the fuzzy string search algorithm in SymSpell, texts are cleaned up to improve the detection of keywords which is used to classify whether an image is compliant or not.

4.4 Validation Framework

An object-oriented approach has been built to create a validation framework to make the solution scalable. The custom library comprises of three classes, with methods in each class catering to specific functionality relevant to the classes. Figure 4.2 shows a schematic diagram of the architecture of the validation framework.

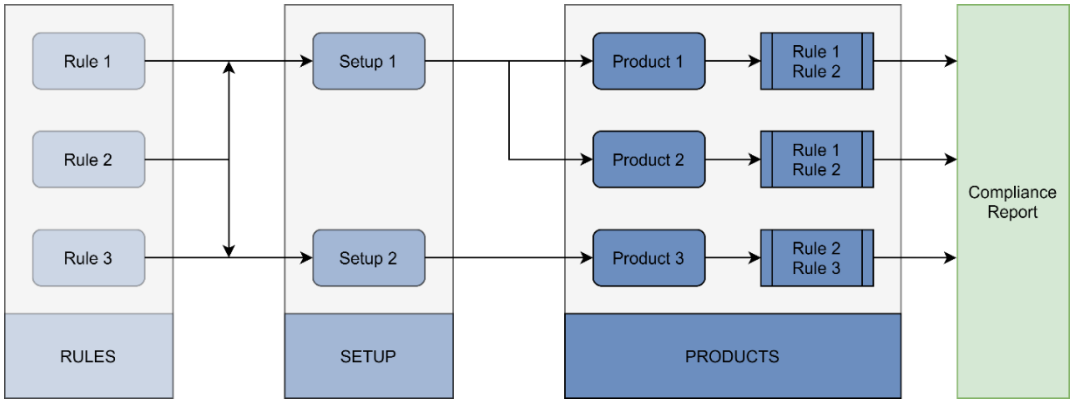

Figure 4.2 Validation Framework Architecture

4.4.1 Rule Class

This class stores the corpus of keywords and the weightage of each keyword belonging to the rule. It also stores the various threshold values for the rule. It has a method defined to calculate and return the score for an image. A dictionary has been used to store the keyword and weightage value pair to improve the time complexity of the searching logic.

4.4.2 Validation Setup Class

This class stores information about the rules that are to be checked for a setup. Many products would require the same set of rules to be checked. In order to eliminate the effort of defining rules to be checked for each product, the concept of creating a setup was introduced. There are two attributes in this class. First, a list which stores the name of the rules to be checked for the products marked to this set-up. Secondly, a switch that enables either of the two object detection options available as mentioned in section 4.2 earlier.

4.4.3 Products Class

This class maps the setup to a product. It also calculates the cumulative score for each relevant rule for the set-up and stores the calculated values. The output to the compliance report is controlled by attributes of this class.

4.5 Implementation of the Framework

The following are the three excel files which have been used to implement the framework:

1. Rules Corpus – Each sheet in this workbook holds the keywords and the weightage for each keyword. An object of the ‘Rule’ class is created for each tab, with the name of the tab being the name of the rule object. The last sheet holds the various threshold values for each rule. The data from this sheet is used to define the threshold values for each rule object created.
2. Setup – This workbook has only one tab. The setup name along with the rules is given. All possible rules are present as column names, with the ones relevant for each setup marked as ‘1’ and the others marked as ‘0’. This sheet is used to create objects of the ‘Validation Setup’ class and attach relevant ‘Rule’ objects to the setup object.
3. Product Hierarchy – This workbook holds the information about each product. The relevant setup for each product is controlled by a hierarchical structure. A suitable hierarchy level can be decided, and the hierarchy code then becomes the setup for the product. Objects of the ‘Product’ class are created, and the relevant setup object is attached, thus integrating the entire framework.

5. MODELS

Primarily, based on previous research, two deep learning models have been used, namely Connectionist Text Proposal Network and the Long Short Term Memory Network.

5.1 Connectionist Text Proposal Network

The CTPN utilizes convolutional layers to identify bounded boxes of relevance and a bi-directional LSTM architecture to derive meaningful text from highly ambiguous texts that are detected by the convolutional layer. An RNN layer, in the end, scores the text output, and texts with positive scores determine whether the current bounded box is of relevance or not. Figure 5.1 shows the typical architecture of a CTPN model. The main advantage of this model is the effective extraction of object information even with a high degree of ambiguity in the texts present on images and thus provides a key advantage to the solution.

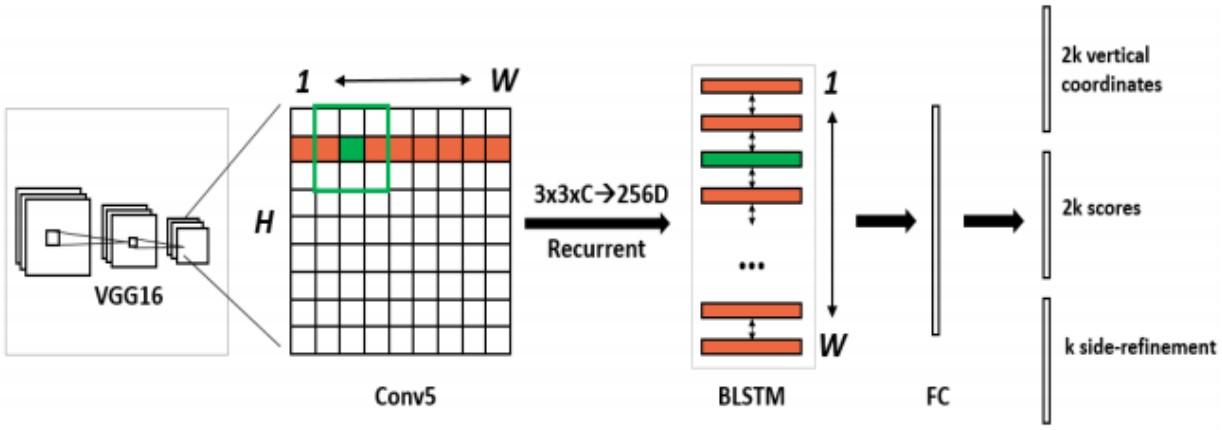

Figure 5.1 A typical Connectionist Text Proposal Network Architecture [16]

5.2 Long Short Term Memory (in Pytesseract)

Pytesseract with LSTM setup has been used in the OCR engine. A gated cell approach has been preferred to avoid the problem of vanishing gradient. Figure 5.2 shows the structure of a single cell in an LSTM network. LSTM is an improvement on a recurrent neural network with the addition of a forget gate in each cell that controls what the cell remembers in a sequenced input. Implementation of LSTM in tesseract has significantly improved its performance in terms of accuracy.

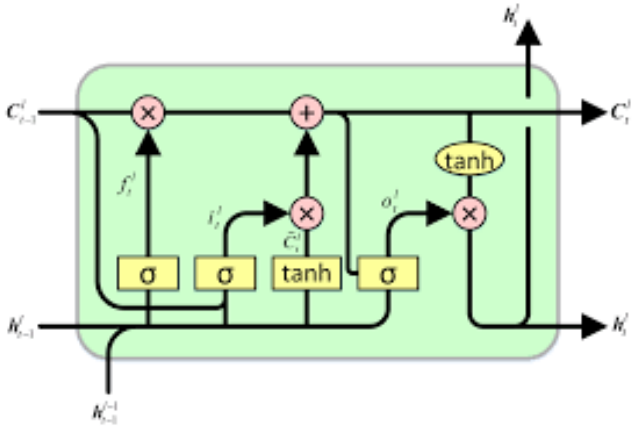

Figure 5.3 An LSTM cell

6. RESULTS

The performance metric of the solution and the various degrees of compliance that the current products hold are the key outcomes of this solution and have been discussed below.

6.1 Performance metric

In the problem being solved, considering the cost of misclassification, a false positive holds a very high cost. This is because, if a product is wrongly marked compliant, then it poses legal risks to the company. Sensitivity is also an important factor as well since false negatives will lead to manual intervention to check for compliance which results in an unnecessary effort. Figure 6.1 shows the confusion matrix for the three categories of labels checked in this experiment along with the specificity and sensitivity achieved for each case.

Figure 6.1 Confusion Matrix for each category of labels

6.2 Computation performance

Keeping scalability in mind, computational efficiency is key. Figure 6.2 gives a bird's eye view of the average time taken for each process in the pipeline category wise. It has been observed that text extraction using the OCR engine and text processing took more time in product categories which had nutrition labels to be detected. For other categories, exception handling took a major time. The compliance check using dictionaries took the least time as expected because of implementation using hash tables.

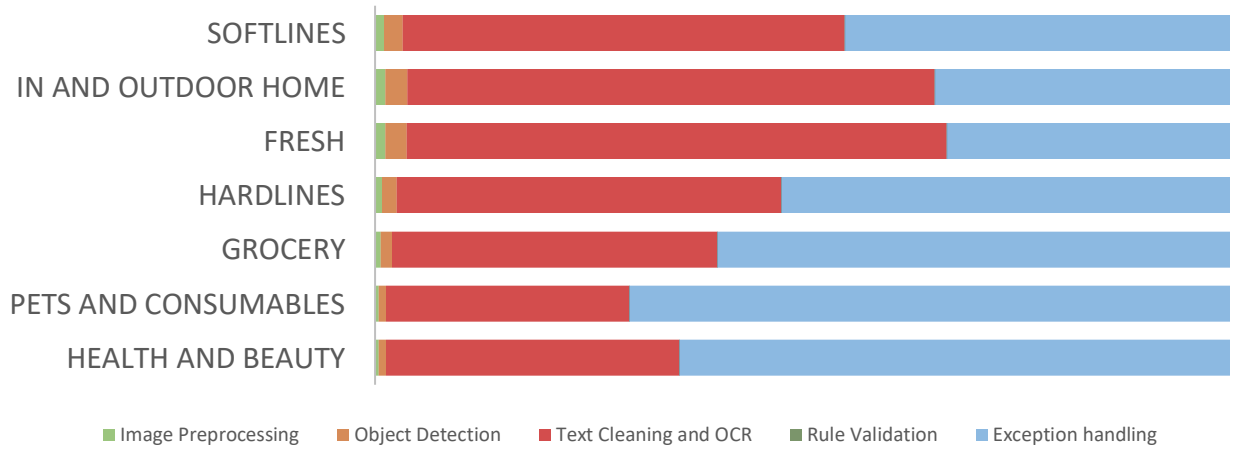

Figure 6.2 Comparison of computational performance

6.3 Compliance check results

The key outcome of this project was to indicant non-compliant products present on the digital platform. Taking a rule wise view of the results in figure 6.3, we can observe that overall, most products which required to have a drug fact label had them. Nutrients and ingredients information was missing for most of the products which required such information. Hence, there needs to be a further investigation concerning the operational or communication issues that might exist between the vendors and the company to improve the compliance statistics.

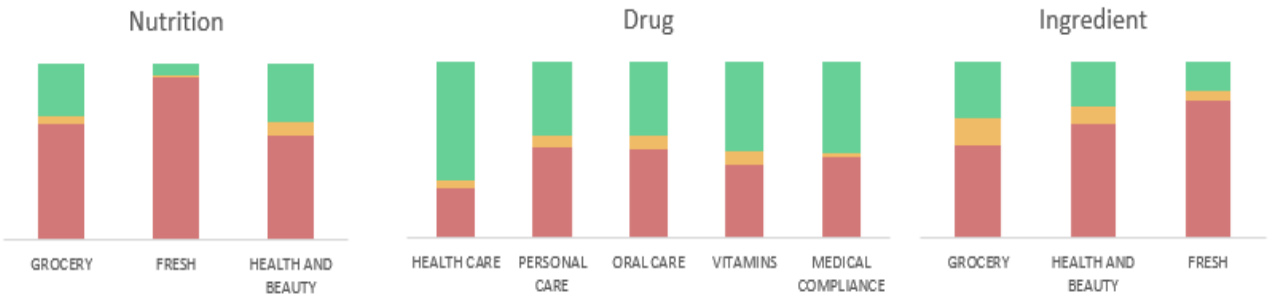

Figure 6.3 Category wise compliance check results

7. CONCLUSIONS

As mentioned in section 2, violations of ADA compliance can lead to hefty fines. Hence, a high standard quality check is of great importance. Automating the audit process has two primary benefits- reduction of manual intervention and standardization of the process. The highly automated audit pipeline, implemented using an object-oriented approach, is scalable and user-friendly. This has a direct impact on resources spent on this process by the firm, reducing it by a staggering 50% indicating a significant operational impact. Also, the high sensitivity achieved by the solution leads to an overall potential savings of $7.5 million for the US retail company. The solution is re-usable in any industry where validation checks need to be done based on information extracted from images.

The following are the scope for some future enhancement to this framework:

- Creation of a user-friendly GUI to feed information that is currently handled using excel files.
- Automate communication based on the output of the compliance check.
- Extend the OCR model to cater to other languages.
- Enhance the solution to cater to video feeds which can be used in manufacturing lines

8. REFERENCES

1. Anand V. Saurkar, Kedar G. Pathare and Shweta A. Gode, 2018. An Overview On Web Scraping Techniques And Tools. *International Journal on Future Revolution in Computer Science & Communication Engineering*, http://www.ijfrcsce.org/download/browse/Volume_4/April_18_Volume_4_Issue_4/1524638955_25-04-2018.pdf
2. Wojciech Bieniecki, Victor Manuel Martínez González and Szymon Grabowski, 2007. Image Preprocessing for Improving OCR Accuracy. *IEEE Conference Publication*, https://ieeexplore.ieee.org/abstract/document/4283429
3. Dan S. Bloomberg, Gary E. Kopec and Lakshmi Dasari, 1995. Measuring Document Image Skew and Orientation. *International Society for Optics and Photonics*, www.spiedigitallibrary.org/conference-proceedings-of-spie/2422/0000/Measuring-document-image-skew-and-orientation/10.1117/12.205832.short?SSO=1.
4. T. Zaman and V. Kulyukin,2015. Text Skew Angle Detection in Vision-Based Scanning of Nutrition Labels. *Int'l Conf. IP, Computer Vision and Pattern Recognition*, https://pdfs.semanticscholar.org/0ce5/a89f1bda4459d52154aec03367b907a2b6f2.pdf
5. Zaman, Tanwir,2016. Vision Based Extraction of Nutrition Information from Skewed Nutrition Labels. *DigitalCommons@USU*, https://digitalcommons.usu.edu/etd/4893/
6. Nate Matsunaga and Rick Sullivan,2015. Image processing for the extraction of nutritional information from food labels. *Scholar Commons, Santa Clara University*, https://scholarcommons.scu.edu/cgi/viewcontent.cgi?article=1041&context=cseng_senior
7. M. Zahid Hossain, M. Ashraful Amin and Hong Yan, 2012. Rapid Feature Extraction for Optical Character Recognition, *Cornell University*, https://arxiv.org/abs/1206.0238
8. Zhang, Xiang, Junbo Zhao and Yann LeCun, 2015. Character-Level Convolutional Networks for Text Classification. *NIPS*, http://papers.nips.cc/paper/5782-character-level-convolutional-networks-for-text-classifica.
9. Khasgiwala, Anuj,2018. Word Recognition in Nutrition Labels with Convolutional Neural Network. *DigitalCommons@USU*, https://digitalcommons.usu.edu/etd/7101/
10. M. R. Gaikwad and N. G. Pardeshi, 2016. Text Extraction and Recognition Using Median Filter. *IJRET*, https://pdfs.semanticscholar.org/6e67/c81ec0db00efad38de77006a33a3000f89af.pdf
11. Parisa Pouladzadeh, Shervin Shirmohammadi and Rana Almaghrabi, 2014. Measuring Calorie and Nutrition from Food Image. *IEEE*, https://ieeexplore.ieee.org/abstract/document/6748066
12. Sefer Kurnaz and Yunus Ziya Arslan, 2018. Object-Oriented Programming in Computer Science. *IGI Global*, https://www.igi-global.com/chapter/object-oriented-programming-in-computer-science/184444
13. V.R. Kanagavalli and G. Maheeja, 2016. A Study on the usage of Data Structures in Information Retrieval. *Cornell University*, https://arxiv.org/abs/1602.07799
14. Cabral, Bruno, and Paulo Marques, 30 July 2007. Exception Handling: A Field Study in Java and .NET. *SpringerLink, Springer, Berlin, Heidelberg*, https://link.springer.com/chapter/10.1007/978-3-540-73589-2_8
15. Cabral, Bruno, and Paulo Marques, 30 July 2007. Exception Handling: A Field Study in Java and .NET. *SpringerLink, Springer, Berlin, Heidelberg*, https://link.springer.com/chapter/10.1007/978-3-540-73589-2_8
16. Zhi Tian ,Weilin Huang,Tong He,Pan He and Yu Qiao. Detecting Text in Natural Image with Connectionist Text Proposal Network. https://bestsonny.github.io/resources/ztian2016_eccv.pdf