Detecting, Classifying, and Mapping Retail Storefronts Using Street-level Imagery

Shahin Sharifi Noorian, Sihang Qiu, Achilleas Psyllidis, Alessandro Bozzon and Geert-Jan Houben
Delft University of Technology, Delft, Netherlands
{s.sharifinoorian,s.qiu-1,a.psyllidis,a.bozzon,g.j.p.m.houben}@tudelft.nl

Abstract

Up-to-date listings of retail stores and related building functions are challenging and costly to maintain. We introduce a novel method for automatically detecting, geo-locating, and classifying retail stores and related commercial functions, on the basis of storefronts extracted from street-level imagery. Specifically, we present a deep learning approach that takes storefronts from street-level imagery as input, and directly provides the geo-location and type of commercial function as output. Our method showed a recall of 89.05% and a precision of 88.22% on a real-world dataset of street-level images, which experimentally demonstrated that our approach achieves human-level accuracy while having a remarkable run-time efficiency compared to methods such as Faster Region-Convolutional Neural Networks (Faster R-CNN) and Single Shot Detector (SSD).

Keywords

Urban Data Extraction; Street-level Imagery; Convolutional Neural Networks; Object Detection

ACM Reference Format:
Shahin Sharifi Noorian, Sihang Qiu, Achilleas Psyllidis, Alessandro Bozzon and Geert-Jan Houben. 2020. Detecting, Classifying, and Mapping Retail Storefronts Using Street-level Imagery. In *Proceedings of the 2020 International Conference on Multimedia Retrieval (ICMR '20), June 8–11, 2020, Dublin, Ireland.* ACM, New York, NY, USA, 7 pages. https://doi.org/10.1145/3372278.3390706

1 Introduction

Commercial facilities are integral to cities worldwide. Listings of these facilities are used in mapping and location services, recommendation systems, search engines, and social media platforms. For these systems to provide accurate and reliable information to the users, it is important that such listings are kept up to date. One of the most challenging issues in this regard is keeping track of the frequent changes that characterize this type of businesses (e.g. a candy shop turning into a bakery).[1] It is estimated that 10% of establishments go out of business every year, and in some market segments (e.g. restaurants), the rate is as high as 30% [31].

The traditional way of keeping such listings up to date requires lots of manual work and often also entails the integration of several third-party resources (e.g. data from the local chamber of commerce). An opportunity to complement these conventional approaches arises from the recent advent of street-level images, available on a variety of platforms (e.g. Google Street View or Mapillary). These frequently updated panoramic views of the urban environment allow to retrieve pictures of the storefronts at large scale. We argue that the information included in the storefronts (e.g. commercial logos, names, text etc.) could help identify the type of business establishment. Recent studies have used street-level imagery in analyzing various aspects of the urban environment [1, 4, 11, 47], and in automatically detecting urban objects [1, 33].

In automatically *detecting*, *mapping*, and *classifying* commercial functions from street-level imagery, the most challenging aspects are: (1) the high degree of visual variability that characterizes storefronts, and that hinders the achievable accuracy and generalizability of prediction models; (2) image acquisition factors such as noise, motion blur, occlusions, lighting variations, specular reflections, perspective, and geo-location errors; and (3) the need for methods with good runtime execution performance, given the continuous changes and the large number of businesses in a city.

This paper presents a multi-modal late-fusion method that combines visual and textual cues in street-level images, and is able to resolve semantic ambiguities and incorrect digitization of detected textual labels. This method detects the physical extent of storefronts, identifies their corresponding commercial function, and approximates their geo-location. Thanks to this late-fusion approach, the recognition module can easily be configured to work on different datasets (e.g. storefronts from different countries) with improved modularity and minimal visual model re-training.

We compare our approach with two state-of-the-art methods, Faster R-CNN [37] and SSD [23], which have shown superior performance in several object detection challenges [22, 28]. Results show that, while having higher precision than Faster R-CNN (2%) and SSD (9%), our approach is considerably faster than the baselines (up to 60%). Furthermore, we show that our proposed classification method is able to outperform state-of-the-art computer vision approaches for POI classification – Places365-CNNs [45] – by 16.86%, and multimodal approaches – Karaoglu et al. [14] by 6.8%. Finally, we ran a crowd-sourcing campaign on Amazon Mechanical Turk, and show that our proposed approach achieves almost the same precision and recall as a human annotator, on detecting and classifying retail storefronts. We also investigate the performance of our method in a number of edge cases, to highlight limitations and suggest future directions of improvement.

[1] Examples, the business-related place types taxonomy, code and dataset are available on the companion page: https://sites.google.com/view/storefrontsmapping

ICMR '20, June 8–11, 2020, Dublin, Ireland
© 2020 Association for Computing Machinery.
ACM ISBN 978-1-4503-7087-5/20/06…$15.00
https://doi.org/10.1145/3372278.3390706

2 RELATED WORK

In this section we discuss related work on knowledge extraction using street-level imagery and fine-grained scene classification.

Knowledge Extraction using Street-level Imagery. Street-level imagery can be a useful data source to extract knowledge about the urban environment [33], especially for tasks requiring high spatial coverage. Recent work shows the feasibility of utilizing street-level imagery in assessing structural changes in urban areas [1], inferring subjective properties of urban areas such as safety, liveliness, and attractiveness [8], mapping urban greenery [18–20], geo-locating high-density urban objects [33], or estimating city-level travel patterns [11]. Other works applied computer vision techniques to Google Street View images for inferring the socioeconomic attributes of neighbourhood in US [9], finding morphological characteristics to distinguish European cities [7], detection of building entrance in outdoor scenes [40], or detection and classification of traffic signs [3]. Yu et. al. [43] addresses the problem of detecting storefronts using street-level imagery. The authors trained a deep learning model on a proprietary dataset (∼ 2M annotated images), however, without addressing the issue of classification into business-related categories. To the best of our knowledge, our work is the first to address the problem of storefront detection, classification, and geo-localization in an integrated fashion.

Fine-grained scene recognition. Alternatively, scene Deep Convolutional Neural Networks (CNNs) have been successful in various scene recognition tasks. However, such breakthroughs in scene recognition do not imply that these models are suitable for fine-grained classification of storefronts based on their visual appearance. This is due to the high degree of intra-class and the low degree of inter-class differences in the appearance of store fronts across business categories [14]. Yan et al. [42] take Spatial Context (i.e. nearby places) into account as complementary information to boost the performance of CNN models for classifying business places. In addition, fine-grained classification based on textual components (i.e. signs, logos etc.) could play an important role in making more accurate predictions [27]. Along these lines, Karaoglu et al. [15] propose a multimodal approach that combines visual features and textual information from imagery into a single feature space as input for a SVM classifier. Our work differs in that we incorporate a multimodal late fusion approach proposed by [29], together with storefront object detection, localization and aggregation algorithms based on street-level imagery, to create an integrated framework for detecting, classifying, and mapping storefronts.

3 Method

Our model consists of three main modules Figure 1. The *storefront detector* extracts the physical extent of retail storefronts from street-level imagery. As there is often more than one storefront in street-level imagery, the detector module outputs a list of bounding boxes. Then detected bounding boxes are iteratively fed into both *classification* and *geo-location estimation* modules. The *classification* module utilizes the bounding box information to crop the original input image, and outputs a probability distribution over candidate classes (business types). The *geo-location estimation* module calculates the actual latitude and longitude of each detected bounding box by using the metadata of street-level imagery.

3.1 Storefront Detection

For fast storefront detection, we rely on the YOLOv3 [36] one-stage object detector. While not being the most accurate object detection algorithm, YOLOv3 suits our requirements as it is a very suitable choice for near real-time detection.

Training. We manually annotated the storefronts in 1200 street-level images. The images were randomly collected from 5 different countries using Google Street View. We divide the dataset into three parts: training (∼ 1000images), validation (∼ 100images), and test (∼ 100 images). We also augment the labeled training data by adding Gaussian noise, varying Brightness, and randomly Rotating image, which results in $1000 \times 5 = 5000$ images in the training set. Due to scarcity of well-annotated data for business storefront detection, as previous studies suggested [30, 41], we use a Transfer Learning strategy in order to improve the quality of our detector. As the designer of YOLO has already pre-trained the network using the OpenImages dataset to extract features [17], we immediately applied the pre-trained weight values for further training.

Due to the visual ambiguity of the storefronts that are too far from the image acquisition device, we remove from the original YOLOv3 architecture the 12 layers that are responsible for detecting very small objects, also obtaining a decrease in the training and inference time of ∼ 10010%. Details on the neural network architecture are provided on the companion page.

The input training images are resized to 416×416, and the network has been trained for 5,000 iterations with batch size of 64. At the end of the training, the loss converges to less than 0.04 on the validation set. As Georgakopoulos et al. [10] suggested, at a general improvement for training process, we initially set the learning rate to 0.001 for the first 3,000 iterations as we are starting with zero information and so the learning rate needs to be high. After 3,000 iterations, we decrease the learning rate to few steps by a factor of 0.1. The YOLO network predicts bounding boxes using dimension clusters, called anchor boxes [35]. We calculated anchor boxes for our storefront dataset using the k-Means algorithm and adapted in our output layers.

Inference. At the inference stage, the final output is delivered in shape of a storefront box, paired with its corresponding confidence score. Given a 416×416 image, our storefront detector outputs $(13 \times 13 + 26 \times 26) \times 3 = 2,535$ bounding boxes. Boxes are filtered on their objectness score, i.e. how likely the box contains an object [36]. Boxes having scores below a threshold are eliminated. Furthermore, we perform Soft Non-maximum Suppression [5] to eliminate redundant overlapping boxes with lower confidences.

3.2 Storefront Classification

The information contained in street-level imagery is essentially visual. Therefore, storefronts can be described based on morphological characteristics (e.g. height, color, materials, geometry) of their facades. Common features which are often found on business-related storefronts, are *signs* or visual *labels*. These contain the name, logo, and other related information that help people identify businesses while navigating through physical space. Therefore, they could be a valuable source of information to classify retail storefronts. Due to the fact that both visual and textual features are important cues, we propose a *multi-modal* approach.

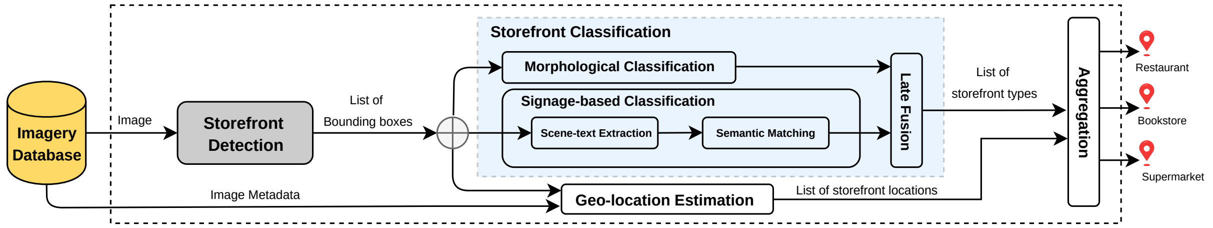

Figure 1: The architecture of our end-to-end model

Morphological Classification. The single-pass object detectors such as YOLO are capable of detecting and classifying objects simultaneously. Due to the high intra-class visual variability (e.g. Italian restaurant and Japanese restaurant) and the small inter-class visual differences (e.g. Bakery and Pizzeria), these models require a large training set containing fine-grained labels assigned to each bounding box. However, collecting such a training set can be very expensive. To address this issue, we separated the visual-based identification of storefront types and formulated it as a scene classification problem. There are several CNN models available for scene classification tasks. As the model selection and comparison are not the focus of our study, we simply employed the Residual Network (ResNet) in our framework, which has shown superior performance on ImageNet classification [12]. We use the pre-trained `ResNet152-places365` model provided by [45], which outputs 365 scene classes, including some (e.g. `cliff` or `coral`) that do not qualify as POI type (e.g. `restaurant`) and, therefore, are not relevant in our setting. Without losing generality, we select 24 business-related place types as our candidate class labels. The list of business-related place types is available on the companion page.Then, we fine-tune the pre-trained `ResNet152-places365` with a 24-way softmax classifier on the subset of the *Places* dataset. The last fully-connected layer of the pre-trained network was removed and replaced by a new fully-connected layer with 24 nodes to solve the storefront type classification problem. Then, the weights of the added fully-connected layers were randomly generated from a Gaussian distribution with zero mean and standard deviation of 0.01.

Signage-based Classification. The first step to identify the storefront type by its signage is to localize and crop textual information in the image in the form of word boxes. However, Scene-text detection is not a trivial task as scene texts have different sizes, width-height aspect ratios, font styles, lighting, perspective distortion, and orientation. There are many scene-text detection methods. As the goal of this work is not to compare different methods/models, we utilize one of the state-of-the-art methods, called CRAFT [2]. The CRAFT model utilizes a convolutional neural network which makes prediction on the character region score and character affinity score. The region score is then used for localization of individual characters on an image, and the affinity score is employed to group each character for constructing a single word instance. We benchmarked the CRAFT model on the *UBER-Text* dataset [44]. It outperforms other state-of-the-art methods such as TEXTBOX++ [21] by 5%, and EAST [46] by 6.5%. We employed the weight values which are pre-trained on the *IC15* dataset [16].

As done with text localization on images, we need to transcribe extracted word (containing rotated, scaled and stretched characters) images into machine-readable character sequences. To this end, we integrate a multi-object rectified attention network (MORAN), proposed by [25], as in several testing rounds MORAN outperformed other state-of-the-art methods on the *UBER-Text* dataset [44].

Next, we integrate a semantic matching approach to predict the type of storefront based on the semantic distance between the words extracted from the image and the standard name of each candidate storefront type, as defined in *ImageNet* synset.[2]

However, some words in street-level imagery may have high similarity with one of the candidate classes, others may be completely irrelevant. For instance, words such as `hair`, `nail` or `beauty` on storefront images are likely to be related to a `Beauty Salon`. On the contrary, `OPEN/CLOSE` signs do not give any information about the type of storefront, which need to be discarded. We therefore remove common – yet irrelevant – words, including verbs like `open`, `close`, `push`, `pull`, etc. using an extended version of the standard stopword list. After reducing potential noise, the recognized word is transformed into a word vector representation. While there can be many implementations for capturing semantic relatedness[24], previous studies have shown that *word embeddings* [26, 32] perform this task particularly well by measuring the cosine similarity of the word embedding vectors. We employ FastText [6] to transform recognized texts into a word vector representation due to its promising performance in overcoming the problem of out-of-vocabulary words, by representing each word as a bag of character n-grams.

As our evaluation will focus on the Manhattan Borough of New York City, the pre-trained (on Common Crawl and Wikipedia [3]) word vectors for English are used. According to the desired language l, the corresponding pre-trained word vector V_l is selected; then, each recognized word is represented by the pre-trained 300-dimensional word vector as v_i. Finally, we use the method proposed by [39] to align the V_l in the same space as the English word vector for multilingual semantic matching. Similarly, each candidate class of storefront type C is represented by a word vector c_j with an English word embedding as reference. Then, we calculate the cosine similarity between each class label (c_j) and each spotted text (v_i) as follows:

$$\cos\left(\Theta_{ij}\right) = \frac{{v_i}^T c_j}{|v_i|\,|c_j|} \tag{1}$$

The probability score P_j for each candidate storefront type (j) is calculated by averaging similarity scores of all spotted words:

$$P_j = \frac{\sum_{i=1}^{K} \cos\left(\Theta_{ij}\right)}{K} \tag{2}$$

[2] http://www.image-net.org/synset
[3] https://fasttext.cc/docs/en/crawl-vectors.html

where $i = 1, 2, ..., K$ is the index of each spotted text, $j = 1, 2, ..., N$ is the index of each probability score, and N is the number of candidate classes.

Then, we utilize a softmax function to normalize the probability scores P_j so that, similar to the morphological classifier, the probability scores of signage-based classifier sum up to 1.

Late Fusion. We use a Linear Bimodal Fusion (LBF) method (here essentially a 2-component convex combination), which linearly combines the ranking scores provided by the CNN model and the semantic similarity scores from the scene-text semantic recognition module, as shown in Equation 3.

$$S_{mixed}(d) = w_v.S_v(d) + w_t.S_t(d) \tag{3}$$

where S_{mixed}, $S_v(d)$, and $S_t(d)$ refer to the final ranking score, visual recognition score, and semantic similarity score for storefront type d respectively, w_v and w_t are the weights for the scene recognition component and scene-text extraction component, and $w_v + w_t = 1$. The weights are determined according to the relative performance of the individual components. Specifically, the weight for the scene recognition module is determined as:

$$w_v = \frac{acc_v}{acc_v + acc_t} \tag{4}$$

where acc_v and acc_t are the measured top@1 accuracy of morphological classifier and signage-based classifier on a manually labeled test set, respectively.

3.3 Geo-location Estimation and Aggregation

To geo-locate the storefronts we propose a storefront geo-location estimation algorithm working on the street-level images metadata. In previous work [33] the geo-location of an urban object is calculated using the intersection of the central line (symmetry line) of the bounding box and the ground-level horizontal plane (i.e. city ground). We adapt [33] by relying on third-party information about existing buildings, and then finding which building facade has an intersection with the given bounding box and then calculating the geo-location of the intersection.

We acquire the data of all the building facades in a city from OpenStreetMap (OSM).[4] The map from OSM is composed by nodes and ways, i.e. points and segments. We extract all the segments having the attribute "building" into a set noted as S, representing the collection of all the building facades.

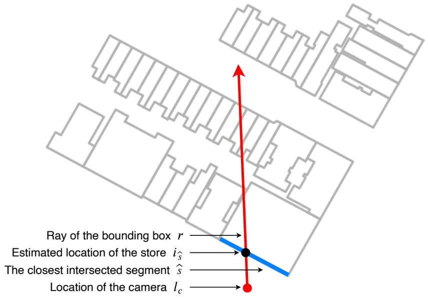
Ray of the bounding box r
Estimated location of the store $i_{\hat{s}}$
The closest intersected segment $\hat{s}$
Location of the camera l_c

Figure 2: Estimation of a storefront's location.

To estimate the geo-location of the storefront (Figure 2), we trace a ray starting from the location of the camera l_c and going with the heading of the bounding box h, where l_c can be immediately acquired from the metadata of the street-level image and h can be easily calculated according to the position of the bounding box on the image [33]. All the segments (facades) close (not farther than R meters) to the camera location l_c are selected into a set S_c ($S_c \subset S$). Then, we check if a segment s ($s \in S_c$) has an intersection with the ray. If the intersection i_s exists, the distance d_{cs} from the camera l_c to the intersection i_s is calculated and recorded. The segment with the minimum d_{cs} is the facade $\hat{s}$ having the targeted storefront. The location of the corresponding intersection $i_{\hat{s}}$ is the estimated geo-location of the storefront.

The same storefront might be annotated multiple times from different street-level images or different crowd workers via either an automatic detection method or crowdsourcing. Therefore, "raw" annotations (bounding boxes with labels) produced by automatic detection or crowdsourcing are aggregated to acquire a single estimated annotation for each storefront. We adopt the density-based location aggregation algorithm proposed by [38], which produces one single estimated geo-location from multiple annotations for each storefront. Based on this, the label with the highest confidence score from candidate annotations is selected as storefront type.

4 Evaluation

In this section, we first describe how datasets are prepared. Then, we separately compare the performance of our detection and classification methods (Shown in Figure 1) with: 1) State-of-the-art approaches; and 2) Human annotators. Finally, we comment on the results of the model (detection, classification, and geo-localization).

4.1 Implementation Details

All the training and experiments are conducted on a NVIDIA Tesla K80 GPU. Our method is not trained in an end-to-end manner. The object detection method is trained using Darknet framework[34] due to its compatibility with the YOLO architecture. To train the other components of our system as well as fine-tuning compared baselines, we use Tensorflow as training platform. We perform all experiments using OpenCV, which provides a generic inference module for various type of Deep Learning models.

4.2 Dataset

We manually annotated 100 street-level images as test set for storefront object detection. The dataset comprises 317 storefront bounding boxes ($\sim$ 3.2 boxes per image). We refer to this dataset as *Store-Obj*. We also collected a set of single storefront images, which are manually classified into 24 categories.The list of categories comprises 24 top business types, which are ranked based on their occurrence in the popular business listings such as Yelp [5] or Foursquare[6]. We name this dataset as *Store-Scene*. All images in *Store-Scene* only contain single store, while *Store-Obj* comprises complex panorama images including more than one storefront, and many more irrelevant urban objects. Ultimately, we created a small benchmark

[4] https://www.openstreetmap.org/

[5] https://www.yelp.com/

[6] https://foursquare.com/

dataset in Manhattan, New York City. We selected a street of about three kilometers long and iteratively collected 150 panoramic images along the street using Google StreetView API. Then, we manually verified that ∼ 120 unique businesses exist in the vicinity of the street mentioned above using Google Places API[7]. We observed that the type of collected businesses corresponds to 18 categories of *Store-Scene* and other 6 business categories do not exist in the area of our experiment. This dataset is used for evaluating the entire pipeline of our model (detection, classification, and geo-localization) in comparison to human performance, as well as for qualitative analysis. We refer to this dataset as Store-location. The properties of each dataset are described in Table 1.

Table 1: Dataset statistics: D = Detection, C = Classification, G = Geo-localization

Dataset	Problem	#Categories	Training	Validation	Testing
Store-Obj	D	1	1,000	100	100
Store-Scene	C	24	-	-	1,100
Store-Location	D, C, G	18	-	-	150

Table 2: Results of our store-front detector method and the state-of-the-art methods with regard to recall (%), precision (%), F1 score, mean average precision over 0.5 IoU threshold (%), and inference time per image (ms).

Method	Recall	Precision	F1 score	mAP@0.50	Infer. time
SSD	68.29	72.35	70.26	74.3	220
Faster R-CNN	**77.03**	79.33	**78.16**	78.9	325
Ours (yolo-storefront-416)	74	**81**	77	**79.37**	**100**
Ours (yolo-storefront-608)*	89.05	88.22	88	91.35	175

∗ The second variation of our model is presented to show the impact of input size.

Table 3: Results of our storefront classification method and the state-of-the-art methods with regard to top@1 accuracy (%), top@5 accuracy (%), and inference time per image (ms).

Dataset	Method	Top@1 acc.	Top@5 acc.	Infer. time
Store-Scene	GoogLeNet-places365	21.45	55.42	**95**
	ResNet152-places365	28.15	59.45	125
	Karaoglu et al. [14]	38.17	69.56	110
	Ours (textual-only)	42.55	80.44	190
	Ours (visual + textual)	**45.01**	**89.44**	205

4.3 Comparison with Object Detectors

We compare our proposed store-front detection approach with Faster R-CNN[37] and Single Shot Detector[23], and measure precision, recall, F-score, mean average precision over 0.5 IoU threshold, and average inference time per image. Both of the baseline methods have shown superior performance in many general object detection challenges[13]. Therefore, these method are suitable for evaluating our object detection approach. We first fine-tune both baseline methods using the training set of *Store-Obj* dataset. All training images, used for tuning baselines methods, are resized to (416×416). Then, we perform experiments using *Store-Obj* test set. As shown in Table 3, our detection approach outperforms both baseline methods in precision (∼+2% & ∼+9%) and mAP@0.5 (∼+0.5% & ∼+5%). In terms of recall, Faster R-CNN performs better, but it has higher inference time (∼+300%) comparing to our method. In addition, we present second variation our model with a larger input sizes (608 × 608). Increasing input size improves precision and recall by ∼+7.8% and ∼+15%, respectively. However, the average inference time also increases by 75%.

4.4 Comparison with Scene Classifiers

We formulate the identification of store-front type as a fine-grained scene classification problem. We compare the performance of our approach with two visual-only scene recognition baselines on the *Store-Scene* dataset described in Table 1. We fine-tuned both baseline methods in the same way as explained in the Section 3.2. This comparison aims at showing the benefit of leveraging textual information from imagery. As shown in Table 3, our scene classification approach outperforms both visual-only baselines. Results suggest that by considering textual information visible on the outdoor appearance of storefronts, it is possible to achieve high performance also with limited training data.

We also compare the performance of the our classification approach with Karaoglu et al.[14], the best performing state-of-the-art method that addresses the problem of business type classification by leveraging textual information from images. The CNN models, used in this method for visual feature detection, are fine-tuned in the same way as our morphological classifer. Our proposed classification approach outperforms the state-of-the-art top@1 from 38.17% to 45.01% (∼+6.8%) on the *Store-Scene* dataset; and improves the Top@5 accuracy from 69.5% to 89.4% (∼+20%).

4.5 Comparison with Human Annotators

To further assess our model, we also conducted a crowdsourcing experiment through Amazon Mechanical Turk[8]. In the crowdsourcing task, workers are asked to draw a bounding box around every visible storefront on the image, and then, choosing its corresponding category from a given list of 24 business types (Described in Section 4.2). We also added an OTHER category which stands for *unknown* or *not-in-the-list* situations. Each image is annotated by at least three unique human annotators. By tracking `workerId`, the back-end system running on our server ensures that each worker submits at most three tasks to avoid biases due to over-repeated participation.

We published 645 HITs and 318 unique workers executed our tasks. We manually check all the HITs and exclude invalid assessments. The aggregated geo-location of annotations, made by crowd workers, are estimated based on the method explained in Section 3.3. We run our model on the *Store-Location* dataset, the same set of street-level images (resolution: 2000 × 640) as used in the crowdsourcing experiment. Then, we removed duplicate geo-locations from the list of the detections resulting in 97 unique businesses. Each storefront bounding box $\hat{B}$ predicted by our model, is considered as True Positive, if there are at least two bounding boxes B, obtained from crowd-sourcing task, where IoU (Intersection over Union) between $\hat{B}$ and B is greater than 0.5. When $\hat{B}$ is confirmed as True Positive, we compare the result of our storefront classifier with the human categorization. Given L is a set of labels, which are assigned to a storefront bounding box by at least 3 human annotators.

[7] https://cloud.google.com/maps-platform/places/

[8] https://www.mturk.com/

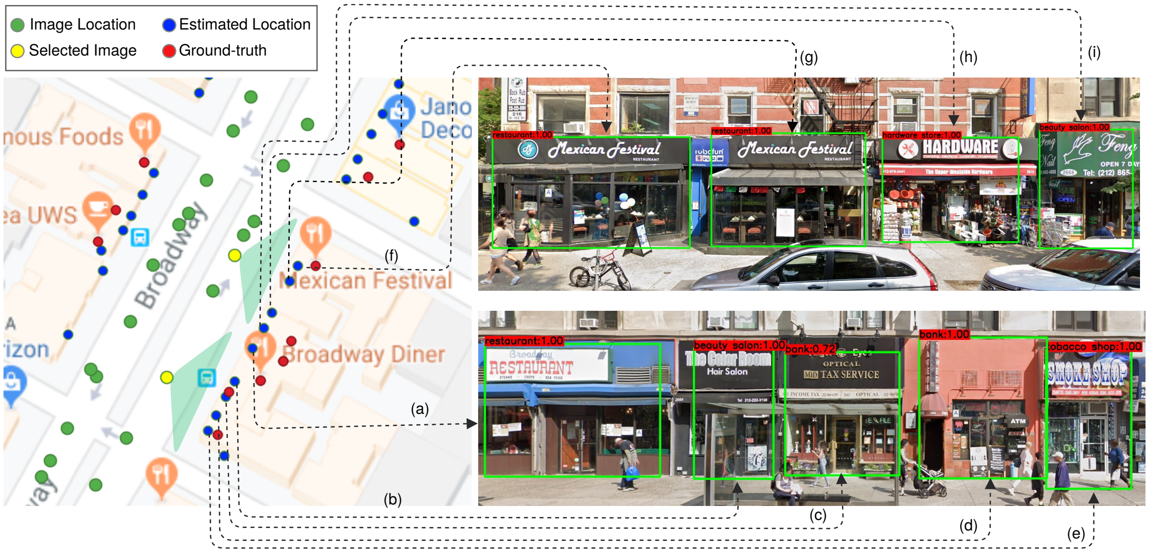

Figure 3: Example applications of our detection, classification, and mapping method.

The predicted set of labels $\hat{L}$ are sorted by classifier's confidence. Then, we define the top k prediction set $\hat{L}_k$ as the first k elements in $\hat{L}$, where $k \in \{1, 5\}$. The prediction of business category $\hat{L}_k$ is confirmed as True Positive, if one label of $\hat{L}_k$ is agreed by at least two human annotators, represented by L. If the best confidence score of top-k predictions is below 0.4, the label is considered as unknown, which is represented by `OTHER` on the list of business categories. As depicted in Table 4, our automatic method achieved 83.2% precision on detecting storefronts: it got 39 false positives out of the 232 detections. Then, we manually removed duplicate geo-locations from the list of the detections resulting in 60 unique businesses. It means a 61.9% recall at 83.2% precision:60 out of 97 businesses visible on Street View imagery were correctly detected.

Table 4: Comparison to human assessment.

	Detection		Classification		Geo-location Estimation	
	Precision	Recall	Top@1 acc.	Top@5 acc.	Precision	Recall
Ours (end-to-end)	83.2%	61.9%	69.1%	92.5%	83.18	61.85

4.6 Qualitative Analysis

We discuss examples of real world scenarios, where the proposed approach provides correct and incorrect predictions on *Store-Location*. Figure 3 shows that our model is able to correctly detect (~89%), classify (~78%), and geo-locate (~89%) business-related storefronts, which are visible in the street-level images. In this example, the storefront (i) is predicted correctly, even when there is no word having direct relation to their types (e.g. *beautysalon*); the proposed semantic matching approach is able to infer that texts such as *Hair* or *Nail*, are semantically close to *beauty salon* in the word vector space, thus enabling correct classification.

A limitation of our system is the difficulty to identify the correct extent of storefronts that are divided into different parts. As Figure 3 shows, the storefronts f and g are detected separately, however, those bounding boxes belong to the same storefront. As discussed in Section 3.2, due to high degree of visual variability, it can be very challenging (if not impossible) to correctly classify the business type of storefronts only based on the visual features. Storefronts (c) and (d) are classified as *Bank*, while the correct labels are *Optician* and *Bar*, respectively. The error for storefront (d) is accountable to the ATM machine sign on the facade, which is the only textual feature, our model is able to extract from the image. As the word 'ATM' usually appear in the same context as 'Bank' in text corpus, our word-vector based semantic-matching method made a wrong prediction with a very high confidence. These failures show an obvious limitation of our method, i.e. that the textual feature might be sometimes misleading, which impacts the overall performance of the proposed approach. Without textual information, the system simply relies on visual features.

5 Conclusion

We introduced a novel approach to detect, classify and geo-locate retail storefronts using street-level imagery. Our approach is able to detect the physical extent of storefronts boundaries even when well-annotated training data is limited. The multi-modal storefront classifier predicts business categories near human-level accuracy by measuring the semantic similarity between detected textual information and the candidate business categories, in addition to morphological characteristics of the storefront's view from the outside. The geo-location aggregation method improves the overall performance of the system by removing false positive predictions. In the future we plan to incorporate additional semantically-rich information, such as contextual information and semantic-relationships between objects, which are visible in the street-level imagery. Furthermore, in order to show the scalability of our approach, we plan to extend the scope of our experiments to other cities in non-English speaking countries.

Acknowledgments. This work has been partially funded by the EU Horizon 2020 program under grant agreement No 874724, and in corporation with WIGeoGIS GmbH in Vienna (Austria).

References

[1] Pablo F Alcantarilla, Simon Stent, German Ros, Roberto Arroyo, and Riccardo Gherardi. 2018. Street-view change detection with deconvolutional networks. *Autonomous Robots* 42, 7 (2018), 1301–1322.

[2] Youngmin Baek, Bado Lee, Dongyoon Han, Sangdoo Yun, and Hwalsuk Lee. 2019. Character Region Awareness for Text Detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 9365–9374.

[3] Vahid Balali, Armin Ashouri Rad, and Mani Golparvar-Fard. 2015. Detection, classification, and mapping of US traffic signs using google street view images for roadway inventory management. *Visualization in Engineering* 3, 1 (2015), 15.

[4] Stefano Bocconi, Alessandro Bozzon, Achilleas Psyllidis, Christiaan Titos Bolivar, and Geert-Jan Houben. 2015. Social glass: A platform for urban analytics and decision-making through heterogeneous social data. In *Proceedings of the 24th International Conference on World Wide Web*. 175–178.

[5] Navaneeth Bodla, Bharat Singh, Rama Chellappa, and Larry S Davis. 2017. Soft-NMS–Improving Object Detection With One Line of Code. In *Proceedings of the IEEE international conference on computer vision*. 5561–5569.

[6] Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov. 2017. Enriching word vectors with subword information. *Transactions of the Association for Computational Linguistics* 5 (2017), 135–146.

[7] Carl Doersch, Saurabh Singh, Abhinav Gupta, Josef Sivic, and Alexei Efros. 2012. What makes paris look like paris? *ACM Transactions on Graphics* 31, 4 (2012).

[8] Kaiqun Fu, Zhiqian Chen, and Chang-Tien Lu. 2018. StreetNet: preference learning with convolutional neural network on urban crime perception. In *Proceedings of the 26th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems*. ACM, 269–278.

[9] Timnit Gebru, Jonathan Krause, Yilun Wang, Duyun Chen, Jia Deng, Erez Lieberman Aiden, and Li Fei-Fei. 2017. Using deep learning and google street view to estimate the demographic makeup of the us. *arXiv preprint arXiv:1702.06683* (2017).

[10] Spiros V Georgakopoulos and Vassilis P Plagianakos. 2017. A novel adaptive learning rate algorithm for convolutional neural network training. In *International Conference on Engineering Applications of Neural Networks*. Springer, 327–336.

[11] Rahul Goel, Leandro MT Garcia, Anna Goodman, Rob Johnson, Rachel Aldred, Manoradhan Murugesan, Soren Brage, Kavi Bhalla, and James Woodcock. 2018. Estimating city-level travel patterns using street imagery: A case study of using Google Street View in Britain. *PloS one* 13, 5 (2018), e0196521.

[12] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 770–778.

[13] Licheng Jiao, Fan Zhang, Fang Liu, Shuyuan Yang, Lingling Li, Zhixi Feng, and Rong Qu. 2019. A Survey of Deep Learning-Based Object Detection. *IEEE Access* 7 (2019), 128837–128868.

[14] Sezer Karaoglu, Ran Tao, Theo Gevers, and Arnold WM Smeulders. 2017. Words matter: Scene text for image classification and retrieval. *IEEE Transactions on Multimedia* 19, 5 (2017), 1063–1076.

[15] Sezer Karaoglu, Ran Tao, Jan C van Gemert, and Theo Gevers. 2017. Con-Text: Text Detection for Fine-Grained Object Classification. *IEEE Transactions on Image Processing* 26, 8 (2017), 3965–3980.

[16] Dimosthenis Karatzas, Lluis Gomez-Bigorda, Anguelos Nicolaou, Suman Ghosh, Andrew Bagdanov, Masakazu Iwamura, Jiri Matas, Lukas Neumann, Vijay Ramaseshan Chandrasekhar, Shijian Lu, et al. 2015. ICDAR 2015 competition on robust reading. In *Document Analysis and Recognition (ICDAR), 2015 13th International Conference on*. IEEE, 1156–1160.

[17] Alina Kuznetsova, Hassan Rom, Neil Alldrin, Jasper Uijlings, Ivan Krasin, Jordi Pont-Tuset, Shahab Kamali, Stefan Popov, Matteo Malloci, Tom Duerig, et al. 2018. The open images dataset v4: Unified image classification, object detection, and visual relationship detection at scale. *arXiv preprint arXiv:1811.00982* (2018).

[18] Xiaojiang Li and Carlo Ratti. 2018. Mapping the spatial distribution of shade provision of street trees in Boston using Google Street View panoramas. *Urban Forestry & Urban Greening* 31 (2018), 109–119.

[19] Xiaojiang Li, Carlo Ratti, and Ian Seiferling. 2017. Mapping urban landscapes along streets using google street view. In *International Cartographic Conference*. Springer, 341–356.

[20] Xiaojiang Li, Chuanrong Zhang, Weidong Li, Robert Ricard, Qingyan Meng, and Weixing Zhang. 2015. Assessing street-level urban greenery using Google Street View and a modified green view index. *Urban Forestry & Urban Greening* 14, 3 (2015), 675–685.

[21] Minghui Liao, Baoguang Shi, and Xiang Bai. 2018. Textboxes++: A single-shot oriented scene text detector. *IEEE Transactions on Image Processing* 27, 8 (2018), 3676–3690.

[22] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. 2014. Microsoft coco: Common objects in context. In *European conference on computer vision*. Springer, 740–755.

[23] Wei Liu, Dragomir Anguelov, Dumitru Erhan, Christian Szegedy, Scott Reed, Cheng-Yang Fu, and Alexander C Berg. 2016. Ssd: Single shot multibox detector. In *European conference on computer vision*. Springer, 21–37.

[24] Christoph Lofi. 2015. Measuring semantic similarity and relatedness with distributional and knowledge-based approaches. *Information and Media Technologies* 10, 3 (2015), 493–501.

[25] Canjie Luo, Lianwen Jin, and Zenghui Sun. 2019. MORAN: A Multi-Object Rectified Attention Network for Scene Text Recognition. *Pattern Recognition* (2019).

[26] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013. Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*. 3111–3119.

[27] Yair Movshovitz-Attias, Qian Yu, Martin C Stumpe, Vinay Shet, Sacha Arnoud, and Liron Yatziv. 2015. Ontological supervision for fine grained classification of street view storefronts. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 1693–1702.

[28] Gerhard Neuhold, Tobias Ollmann, Samuel Rota Bulo, and Peter Kontschieder. 2017. The mapillary vistas dataset for semantic understanding of street scenes. In *Proceedings of the IEEE International Conference on Computer Vision*. 4990–4999.

[29] Shahin Sharifi Noorian, Achilleas Psyllidis, and Alessandro Bozzon. 2019. ST-Sem: A Multimodal Method for Points-of-Interest Classification Using Street-Level Imagery. In *International Conference on Web Engineering*. Springer, 32–46.

[30] Sinno Jialin Pan and Qiang Yang. 2009. A survey on transfer learning. *IEEE Transactions on knowledge and data engineering* 22, 10 (2009), 1345–1359.

[31] HG Parsa, Jean-Pierre I van der Rest, Scott R Smith, Rahul A Parsa, and Milos Bujisic. 2015. Why restaurants fail? Part IV: The relationship between restaurant failures and demographic factors. *Cornell Hospitality Quarterly* 56, 1 (2015), 80–90.

[32] Jeffrey Pennington, Richard Socher, and Christopher Manning. 2014. Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*. 1532–1543.

[33] Sihang Qiu, Achilleas Psyllidis, Alessandro Bozzon, and Geert-Jan Houben. 2019. Crowd-Mapping Urban Objects from Street-Level Imagery. In *The World Wide Web Conference*. ACM, 1521–1531.

[34] Joseph Redmon. 2013–2016. Darknet: Open Source Neural Networks in C. http://pjreddie.com/darknet/.

[35] Joseph Redmon and Ali Farhadi. 2017. YOLO9000: better, faster, stronger. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 7263–7271.

[36] Joseph Redmon and Ali Farhadi. 2018. Yolov3: An incremental improvement. *arXiv preprint arXiv:1804.02767* (2018).

[37] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. 2015. Faster r-cnn: Towards real-time object detection with region proposal networks. In *Advances in neural information processing systems*. 91–99.

[38] Alex Rodriguez and Alessandro Laio. 2014. Machine learning. Clustering by fast search and find of density peaks. *Science (New York, N.Y.)* 344, 6191 (jun 2014), 1492–6. https://doi.org/10.1126/science.1242072

[39] Samuel L Smith, David HP Turban, Steven Hamblin, and Nils Y Hammerla. 2017. Offline bilingual word vectors, orthogonal transformations and the inverted softmax. *arXiv preprint arXiv:1702.03859* (2017).

[40] Mehdi Talebi, Abbas Vafaei, and Amirhassan Monadjemi. 2018. Vision-based entrance detection in outdoor scenes. *Multimedia Tools and Applications* 77, 20 (2018), 26219–26238.

[41] Jonti Talukdar, Sanchit Gupta, PS Rajpura, and Ravi S Hegde. 2018. Transfer learning for object detection using state-of-the-art deep neural networks. In *2018 5th International Conference on Signal Processing and Integrated Networks (SPIN)*. IEEE, 78–83.

[42] Bo Yan, Krzysztof Janowicz, Gengchen Mai, and Rui Zhu. 2018. xNet+ SC: Classifying Places Based on Images by Incorporating Spatial Contexts. In *10th International Conference on Geographic Information Science (GIScience 2018)*. Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik.

[43] Qian Yu, Christian Szegedy, Martin C Stumpe, Liron Yatziv, Vinay Shet, Julian Ibarz, and Sacha Arnoud. 2015. Large scale business discovery from street level imagery. *arXiv preprint arXiv:1512.05430* (2015).

[44] Ying Zhang, Lionel Gueguen, Ilya Zharkov, Peter Zhang, Keith Seifert, and Ben Kadlec. 2017. Uber-text: A large-scale dataset for optical character recognition from street-level imagery. In *SUNw: Scene Understanding Workshop-CVPR*.

[45] Bolei Zhou, Agata Lapedriza, Aditya Khosla, Aude Oliva, and Antonio Torralba. 2018. Places: A 10 million image database for scene recognition. *IEEE transactions on pattern analysis and machine intelligence* 40, 6 (2018), 1452–1464.

[46] Xinyu Zhou, Cong Yao, He Wen, Yuzhi Wang, Shuchang Zhou, Weiran He, and Jiajun Liang. 2017. EAST: an efficient and accurate scene text detector. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*. 5551–5560.

[47] Yi Zhu, Xueqing Deng, and Shawn Newsam. 2019. Fine-grained Land Use classification at the city scale using ground-level images. *IEEE Transactions on Multimedia* (2019).