Fine-grained Image Classification and Retrieval by Combining Visual and Locally Pooled Textual Features

Andres Mafla Sounak Dey Ali Furkan Biten Lluis Gomez Dimosthenis Karatzas
Computer Vision Center, UAB, Spain
{andres.mafla, sdey, abiten, lgomez, dimos}@cvc.uab.es

Abstract

Text contained in an image carries high-level semantics that can be exploited to achieve richer image understanding. In particular, the mere presence of text provides strong guiding content that should be employed to tackle a diversity of computer vision tasks such as image retrieval, fine-grained classification, and visual question answering. In this paper, we address the problem of fine-grained classification and image retrieval by leveraging textual information along with visual cues to comprehend the existing intrinsic relation between the two modalities. The novelty of the proposed model consists of the usage of a PHOC descriptor to construct a bag of textual words along with a Fisher Vector Encoding that captures the morphology of text. This approach provides a stronger multimodal representation for this task and as our experiments demonstrate, it achieves state-of-the-art results on two different tasks, fine-grained classification and image retrieval. The code of this model will be publicly available at [1].

1. Introduction

Written communication is arguably one of the most important human inventions that allows the transmission of information in an explicit manner. Moreover, given the fact that text is omnipresent in man made scenarios [41, 20], as well as the implicit relation between visual information and scene text instances, the design of holistic computer vision models for scene interpretation is fundamental.

With the purpose of designing a holistic model, in this work we leverage textual information applied to the problem of fine-grained classification and image retrieval. Fine-grained classification tackles the problem of classifying different object instances that are visually similar and difficult to discriminate. The complexity of this task lies in finding discriminative features which often require domain specific knowledge [29, 42].

[1] http://www.github.com/DreadPiratePsyopus/Fine_Grained_Clf

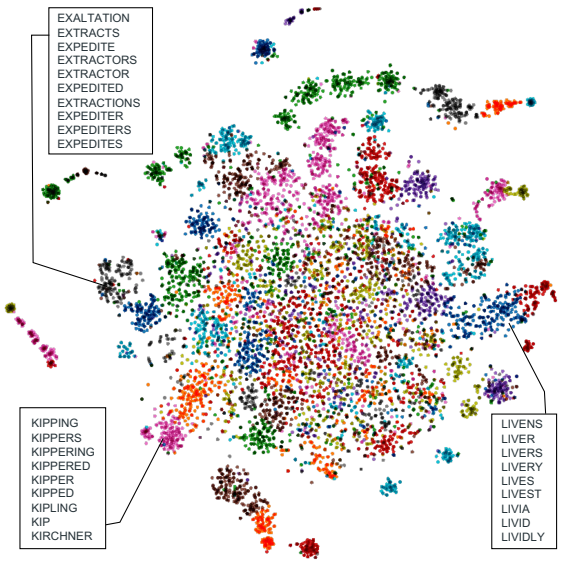

Figure 1. T-SNE Visualization [28] of the 300 dimensional PCAed PHOCs in a two dimensional space. Words with similar morphology are clustered together by a Gaussian Mixture Model, thus making such a descriptor suitable and powerful enough to discriminate text for a fine-grained classification task.

An early work that demonstrated the importance of text (domain specific knowledge) for fine-grained storefront classification was put forward by Movshovitz *et al*. [31], in which the trained classifier learned automatically to attend to text found in an image as the sole way of solving the task. Since then, there has been additional research that explicitly combines textual and visual cues, being the work presented by Karaoglu *et al.* [19, 18] and Bai *et al.*[2] the most related ones to our paper. In this work, we propose the usage of a state of the art text retrieval model presented by Gomez *et al*. [11] to detect and obtain the Pyramidal Histogram of Characters (PHOC) of scene text. We use the PHOC descriptors extracted from images and explore different fusion strategies to merge the visual and textual modalities. Additionally, we construct a Fisher Vector (FV) Encoding

from the obtained PHOCs to be used as a fixed-length text feature in our pipeline and further improve the classifier results. Our model leverages the visual features combined with the morphology of a word (refer to Figure 1), that belong to specific fine-grained classes, without the need to understand them semantically. Contrary to previous methods, this approach is especially useful when dealing with text recognition errors and named entities which are often difficult to encode in a purely semantic space. The combination of these two modalities produce an output probability vector that addresses the classification task at hand. As an additional application, we evaluate the proposed model on fine-grained image retrieval in available datasets. Overall, the main contributions of our work are:

- We propose a novel architecture that achieves state of the art on fine-grained classification by considering text and visual features of an image.
- We show that by using Fisher Vectors obtained from PHOCs of scene text, we obtain a more robust representation in which words with similar structure get encoded on the same Gaussian component, thus creating a more powerful discriminative descriptor than PHOCs alone.
- We provide exhaustive experiments in which we compare the performance of different alternative modules in our model and previous state of the art.

2. Related Work

2.1. Scene Text Detection and Recognition

Even though deep learning has made significant progress [23], localizing and recognizing text in images still remains an open problem in the computer vision community due to the ample variety of text occurrences in natural images [51]. Essentially a system capable of reading text requires two steps, detection and recognition. Jaderberg *et al.* [17] tackles this problem by generating text proposals that were refined by a CNN. The bounding boxes obtained were used as input to another CNN that was trained to classified them according to a fixed dictionary. In another work, [14] defined a Fully Convolutional Regression Network to detect text by regressing bounding boxes and the same classification network as [17] was employed for text recognition. More recent approaches use customized variations of object detectors fine-tuned to detect text instances such as [22] and [27] resulting in models proposed by [50] and [25, 24]. Recently, the community attention has placed an additional effort in the development of end-to-end models. The main existing notion is that features that help to improve detection are also useful at the moment of recognizing text instances. He *et al.* [16] uses a CNN to extract proposals, which are fed into an LSTM (Long-Short Term Memory) to refine the bounding boxes that are later employed as input to yet another LSTM to perform recognition. In parallel, additional work has been conducted into the development of multilingual scene text recognizers, such as the work of [7] which consists on two CNNs. The first one is optimized to detect text and a the second one employs a Connectionist Temporal Classification (CTC) [12] module for recognition, while training both in an end to end manner.

In this work, we leverage the Pyramidal Histogram Of Characters (PHOC) descriptor [1, 38] (see Figure 3) commonly used to query a given text instance in handwritten documents and natural scene images. The PHOC of a word encodes the position of a specific character in a particular spatial region of the detected text instance. Such a descriptor has proven to perform as the state of the art in scene text retrieval [11], and as our experiments show, encoding it with the Fisher Vector [33] provides an improved text descriptor for fine-grained classification.

2.2. Fine-Grained Classification

Recent works on fine-grained classification base their approach on localizing salient parts of an image [9, 44], and use the saliency maps to classify the objects. Later approaches such as the one of Tang *et al*. [40], use a weakly supervised method to find discriminative features and leverage them to perform the classification between similar instances. Other methods use existing prior knowledge from unstructured text to propose a semantic embedding that differentiates similar classes [43]. A self-supervision method is introduced in [45] that learns to propose significant image regions to find inter-class discriminative features.

More related to our work, [18] tackles this task by extracting visual features with a pre-trained GoogleNet [39] and a Bag of Words feature to represent the text instances found in an image and further classify them. More recently, Bai *et al*. [2] use a similar approach and extract visual features using a GoogleNet and a combination of two models: [25] to detect and [35] to recognize text. The text found is represented as GloVe features [32], a word embedding that is further used with attention on the visual features to find a semantic relation between the two modalities to classify the image.

2.3. Multimodal Fusion

The combination of different modalities provides a richer content description rather than one modality alone, therefore the contained knowledge should be leveraged to further exploit explicit information according to the task [37]. In this work we explore other fusion methods used in multimodal learning, that shows a performance increase especially in tasks that require exploiting two modalities such as Visual Question Answering (VQA) and Visual Relationship Detection (VRD).

One of the initial works presented by [3], modeled a

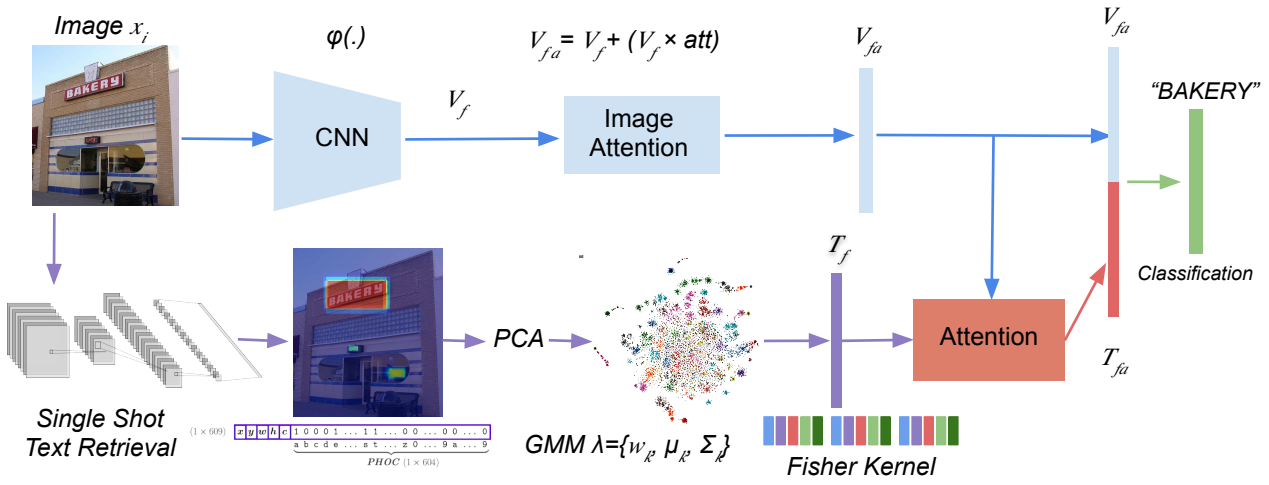

Figure 2. Proposed model pipeline. The PHOCs obtained from [11] are used to compute a Fisher Vector that yields a compact morphology based descriptor suitable to discriminate features from visually similar objects.

Tucker decomposition of the bilinear interaction of two distinct modalities. Later, a Multimodal Low-rank Bilinear Attention Network (MLB) was proposed by [21], in which the result of the fusion of two modalities was based on a low-rank bilinear pooling operation using the Hadamard product along with an attention mechanism. A factorized bilinear pooling (MFB) is proposed by [47], where each third mode section of the tensor is constrained by a rank. Later methods, such as a Multimodal Factorized High-order pooling (MFH) fusion was presented by [48], which uses a high-order fusion formed by cascaded MFB modules. In the work conducted by [3], a bilinear pooling is performed where the tensor is represented as a Tucker decomposition. The obtained main tensor has the same rank constrain as the MFB technique. Lately, a Multimodal Bilinear Superdiagonal Block (Block) fusion strategy based on the work presented by [4], has achieved state of the art results in VQA and VRD.

3. Proposed Model

The devised model consists mainly in four processing blocks: visual features extraction, textual features extraction, attention unit and classification. The whole model pipeline is shown in Figure 2.

The first block extracts the visual features from a given image and produces a fixed size representation of it. The second block consists of extracting the PHOC representation of each text instance found in an image and use a pretrained Gaussian Mixture Model (GMM) to obtain the correspondent FV descriptor. The third block consists of an attention unit that multiplies learned weights with the encoded FV depending on the visual features extracted previously. Finally, the last block consists of a concatenation of the two different modalities followed by a fully connected layer to obtain a probability output vector which is used for classification. For the rest of the paper, let $\mathcal{C}$ be the set of all possible categories in a given dataset; $\mathcal{X} = \{x_i\}_i^N$ be the set of images; $l_x : \mathcal{X} \rightarrow \mathcal{C}$ be the labelling function.

3.1. Visual Features

In our model, we use a Convolutional Neural Network (CNN) [15] pre-trained on ImageNet [8] as a visual feature extractor, denoted as $\phi(\cdot)$. We use the output of the last convolutional block of $\phi(\cdot)$ before the last average pooling layer as the visual features, denoted as V_f. Attention on visual features has proven to yield improved performance on several tasks. As it is presented by [10], we compute a soft-attention mechanism due to its differentiable properties, thus allowing an end-to-end learning. The proposed attention function learns an attention mask att which assigns weights to different regions of an image given a feature map V_f. The attention mask is learned by applying 1×1 convolution layers on the output features from the CNN. Lastly, to obtain the final output of the attention module along with the visual features, the operation is computed by $V_{fa} = V_f + (V_f \times att)$.

3.2. Textual Features

Methods shown in previous works [18, 2] contain mainly three drawbacks. First, the employed text recognizers are bound to a fixed dictionary, which may or may not include the exact words that are present in the image. Second, some words that are contained in the fixed recognition dictionary may not exist in the proposed semantic embedding (GloVe, Word2Vec) such as license plates, brand names, acronyms,

etc. Third, any mistake committed by the recognizer will yield a vector embedding that lies far from the semantic embedding of the correct word. Contrary, correct recognition of semantically similar words that might indicate different fine-grained classes will lead to embeddings close to each other, which are not discriminative enough to perform correct classification. This is the case of similar semantic words such as restaurant and steakhouse, cafe and bistro, coke and pepsi among some other sample classes from the datasets used.

In order to exploit the morphology of a word to obtain discerning features, we employ the PHOC representation. The PHOC representation employed in this work is composed by the concatenation of vectors from the levels 2 to 5 plus the 50 most common bi-grams in English language. This yields a 604-dimensional discrete binary vector that represents the characters contained in a word (see Figure 3). A dictionary given by [17] is employed to obtain a PHOC per word, in this way, we populate a matrix of this compact representation. In order to reduce the dimensionality and to find linearly uncorrelated variables of this compact vector, a Principal Component Analysis (PCA) is performed. This procedure yields a more compact but at the same time informative vectorial representation of a given word.

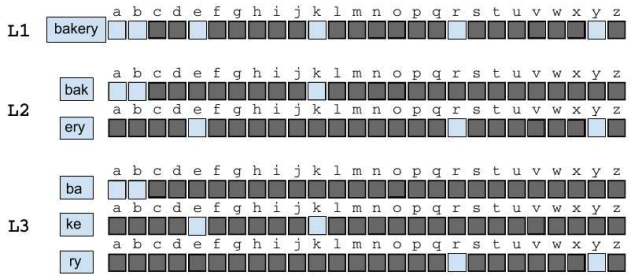

Figure 3. Levels 1 to 3 of the PHOC of the word "bakery". The final compact representation is a concatenation of the histograms of each level. Blues represent "1" while blacks represent "0". Best viewed in color.

The obtained data points were used to construct a Gaussian Mixture Model (GMM) [13] formed by K Gaussian components. We denote the parameters of the K-component GMM by $\lambda = \{w_k, \mu_k, \Sigma_k, k = 1, \ldots, K\}$, where w_k, μ_k and Σ_k are respectively the mixture weight, mean vector and covariance matrix of Gausssian k. We define:

$$u_\lambda(x) = \sum_{k=1}^{K} w_k u_k(x) \tag{1}$$

where u_k denotes Gaussian k:

$$u_k(x) = \frac{1}{(2\pi)^{D/2} \left|\Sigma_k\right|^{1/2}} \exp\left\{-\frac{1}{2}(x-\mu_k)' \Sigma_k^{-1}(x-\mu_k)\right\} \tag{2}$$

and we require:

$$\forall_k : w_k \geq 0, \quad \sum_{k=1}^{K} w_k = 1 \tag{3}$$

Once the GMM model is trained, it will be used to extract a single Fisher Vector representation per image which encodes its contained textual information. The textual features per image are obtained by using the model from [11]. Given an input image, the model outputs a list of $\mathcal{B}$ bounding boxes, each one containing a confidence score $\mathbb{C}$ and a PHOC prediction.

We get the top-m object proposals set $\mathcal{O}_m := \{o \in \mathbb{C}_i : o \geq c, \forall c \in \mathbb{C}_i\}$. The resulting PHOCs $\in [0,1]^{d \times N}$, where d is the dimensionality of the PHOC embedding obtained and N the recognized words embedded in the PHOC space. It is essential to note that the model from [11] is able to generalize and construct PHOCs from previously unseen samples, out of vocabulary words and different languages that employ s similar character set (e.g. Latin), making it suitable for the task at hand. Afterwards, we project each embedded textual instance of the obtained descriptors into a reduced dimensional space by employing PCA. The resulting vectors are used to obtain the Fisher Vector [33] from the previously trained GMM. The GMM associates each PCAed vector o_i to a component k in the mixture model with a weight given by the posterior probability:

$$q_{ik} = \frac{\exp\left[-\frac{1}{2}(o_i-\mu_k)^T \Sigma_k^{-1}(o_i-\mu_k)\right]}{\sum_{t=1}^{K} \exp\left[-\frac{1}{2}(o_i-\mu_t)^T \Sigma_k^{-1}(o_i-\mu_t)\right]} \tag{4}$$

For each mode k, consider the mean and the covariance deviation vectors

$$\begin{aligned} u_{jk} &= \frac{1}{N\sqrt{w_k}} \sum_{i=1}^{N} q_{ik} \frac{o_{ji}-\mu_{jk}}{\sigma_{jk}}, \\ v_{jk} &= \frac{1}{N\sqrt{2w_k}} \sum_{i=1}^{N} q_{ik} \left[\left(\frac{o_{ji}-\mu_{jk}}{\sigma_{jk}}\right)^2 - 1\right] \end{aligned} \tag{5}$$

where $j = 1, 2, \ldots, D$ spans the vector dimensions. The FV of a given image I is simply the concatenation of the obtained vectors u_k and v_k for each of the K components in the Gaussian mixture model.

$$T_f = \left[\cdots \quad \mathbf{u}_k \quad \cdots \quad \mathbf{v}_k \quad \cdots\right]^\intercal \tag{6}$$

The FV and the GMM encode inherently similar information. This takes place because they both include statistics of order 0, 1 and 2 [34, 33]. However, the FV provides a vectorial representation which is more compact, faster to compute and suitable for processing. The dimension of the

FV obtained, noted as T_f, is given by $(2 \times d \times K)$, where d is the PHOC dimension after performing the PCA and K is the number of Gaussian clusters. The intuition captured by the FV is to compute the gradient of a PHOC sample (bag of textual features) that shows the probability of belonging to each of the Gaussian components, which can be understood as a probabilistic textual vocabulary based on its morphological structure (see Figure 1).

3.3. Attention on features

In the proposed fine-grained classification task we can intuitively state that there will be some recognized text that is more relevant than others at the moment of discriminating similar classes. Therefore, it is important to capture the inner correlation between the textual and visual features. To adhere this idea into our pipeline, we propose a modified attention mechanism inspired from [46]. The attention mechanism learns a tensor of weights W that is used between the visual features and the obtained FV. The implemented attention is defined by:

$$W_a = Softmax(tanh(V_{fa}^T \cdot W \cdot T_f)) \tag{7}$$

$$T_{fa} = W_a \cdot T_f \tag{8}$$

The resulting tensor W_a, contains a normalized attention vector that is multiplied with the textual features T_f to obtain the final attended textual features T_{fa}.

The obtained attended textual features T_{fa} and the visual features V_{fa} are concatenated, such that the final features are formed by $F = [V_{fa}, T_{fa}]$. Finally, the resulting vector serves as input to a final classification layer that outputs the probability of a given class. The proposed network is trained to optimize the cross entropy loss function given by:

$$J(\theta) = -\frac{1}{N}\sum_{n=1}^{N}\sum_{i=1}^{C} l_i^n log(\hat{l}_i^n) \tag{9}$$

4. Experiments and Results

The following section describes the datasets employed, the implementation details along with the analysis of the results obtained from the experiments conducted.

4.1. Datasets

4.1.1 Con-Text Dataset

Originally presented by [19], is a dataset taken from the ImageNet [8] "building" and "place of business" subcategories. It consists of 28 categories with $24,255$ images in total. The classes from this dataset are visually similar (Pizzeria, Restaurant, Dinner, Cafe) and requires text to successfully perform a fine-grained classification. The dataset was not built for text recognition purposes, thus not all images contain text in them. A high variability of text size, location and font styles make text recognition on this dataset a challenging task.

4.1.2 Drink Bottle Dataset

Dataset presented by [2] comprises the sub-categories soft drink and alcoholic drink found on ImageNet[8]. There are 18,488 images divided in 20 categories. The dataset contains several not common, occluded, rotated, low quality and blurred text instances which increases the difficulty of performing successful text recognition.

4.2. Implementation Details

The visual features of the proposed model are taken by attending the features of the output of the last block layer of the Resnet152 before the last average pooling layer. These features are passed through a fully connected layer to downsample them to a final dimension of 1×1024. To construct the textual features, a maximum number of $N_{max} = 15$ PHOC proposals are obtained per image. If a lesser number of PHOC proposals are obtained, a zero padding scheme is employed to fix the size of the input features. The resulting PHOCs are reduced in size through PCA, to obtain features of a dimensionality of $N_{max} \times 300$.
The Fisher Vector is calculated from the PCA-ed PHOCs by employing a pre-trained Gaussian Mixture Model as it is described in Section 3.2. The trained GMM employs 64 Gaussian components thus yielding a FV of 1×38400 dimension. The obtained textual features are down-sampled by passing them through a fully connected layer to finally obtain a resulting size of 1×512 before the attention mechanism is computed. The attention between both modalities produces an output vector of 1×512, that multiplies the learned weights to the textual features. As the last step, a concatenated vector of the visual and textual features ($dim = 1 \times 1536$) is used to produce the final classification probability vector.

The network is trained for 30 epochs with the combination of RAdam [26] and the Lookahead [49] optimizers. The batch size employed in all our experiments is 64, with a learning rate of 0.001, momentum of 0.9 that decays by 0.1 every 10 epochs.

4.3. Comparison with the State of the Art

When comparing our method to the current state of the art, it is evident that the proposed pipeline consistently outperforms previous approaches. The performance of our method is shown in Table 1 (see the Supplementary Material Section for the results of each of the classes found in the Con-Text and Drink Bottle datasets respectively). As it can be seen, our method surpasses [2] in the Drink Bot-

tle dataset by a significant margin, however this margin is smaller in the Con-Text dataset. Nonetheless, it is important to note that the method presented by [2] employs two additional classifiers to solve this task, thus relying on an ensemble model. Such kind of adopted approaches require longer training times, as well as more computation resources since several deep networks need to be trained. Therefore, when comparing to the single classifier presented by [2], our model offers a significant improvement. In the upcoming sections, we provide explanations and exhaustive experimentation that shows the main strengths and advantages of our model.

Method	Con-Text	Bottles
Karaoglu[19]	39.0	–
Karaoglu[18]	77.3	–
Bai[2]	78.9	–
Bai*[2]	79.6	72.8
Ours	**80.2**	**77.4**

Table 1. Classification performance for two state-of-the art methods and our proposed model on the Con-Text and Bottles dataset. The results presented by [2] depicted with * are based on an ensemble model.

4.4. Importance of Textual Features

Several baselines of growing complexity were defined in order to: assess the effectiveness of the proposed model, discern the added performance of employing textual features along visual ones and to verify the improvement obtained from using a fusion mechanism.

Visual Only: This baseline assesses the performance of the CNN encoder based on visual features solely. To this end, the 2048 dimensional output features V_f, serve as the input to a fully connected layer according to the number of classes of the evaluated dataset.

Textual Only: We evaluate the performance of two state of the art text recognizers: Textspotter [16] and E2E_MLT [7] along with the most confident PHOCs obtained from the model presented by [11]. For illustration purposes, Figure 4 shows heat maps obtained by employing the model from [11] according to the confidence scores obtained when a text instance is detected. It is important to note that Textspotter [16] is bound to a dictionary to output the final recognized word, whereas the multilingual model E2E_MLT from [7] is not. The recognized text is embedded with pretrained versions of GloVe [32], FastText [6] and Word2Vec [30], finally outputting tensors of size $N_{max} \times 300$, which in our experiments $N_{max} = 15$. When working with PHOCs, the output vector has a size $N_{max} \times 604$. As we can observe in Table 2, in the visual only baseline, the ResNet152 CNN [15] performed better in this task, due to the major expressiveness of the model and the residual block architecture that it is based on.

Figure 4. Heat maps obtained according to the confidence detection score of the predicted PHOCs.

	Model	Con-Text	Bottles
Visual	GoogLenet	61.21	64.93
	Resnet-152	63.70	66.56
Textual	Texspotter+w2v	35.09	50.68
	Texspotter+glove	34.52	50.26
	Texspotter+fasttext	36.71	51.93
	E2E_MLT+w2v	44.36	43.98
	E2E_MLT+glove	44.25	42.64
	E2E_MLT+fasttext	45.07	44.31
	PHOC	**49.18**	**52.39**
	Fisher Vector (PHOC)	**63.93**	**62.41**

Table 2. Visual only and Textual only results. The textual only results were performed on the subset of images that contained spotted text. The metric depicted is the mean Average Precision (mAP in %).

In the text only baseline, by using standard text recognizers we can observe that the E2E_MLT performs better in the Con-Text dataset, whereas the Textspotter model surpasses E2E_MLT in the Drink Bottle dataset. Nonetheless, both of them are outperformed by employing the PHOCs obtained from [11] as the word embedding. This effect is due to the inherent morphological nature of the PHOC embedding.

Overall, the best results in the textual only baseline are obtained by the Fisher Vector obtained from the PHOCs. Qualitatively shown in Figure 1, the Gaussian Mixture gracefully captures the morphology of words obtained from PHOCs. Therefore, words with similar syntax are clustered together in the GMM, thus allowing the Fisher Vector to be a powerful descriptor relevant for this task that yields even more discriminative features than other embeddings. It is important to note as well that in our experiments, Fast-

	Fusion	T+W	T+G	T+F	E+W	E+G	E+F	PHOC	FV(F)	FV(P)
Con-Text	**Concat**	73.84	74.11	74.33	77.04	77.58	77.77	77.45	77.31	**80.21**†
	Block [4]	73.12	73.86	73.18	76.97	78.34	78.34	77.96	77.87	79.27
	Mutan [3]	72.46	72.08	73.47	77.67	77.26	78.05	76.97	76.01	78.51
	MLB [21]	73.17	72.18	74.09	77.45	76.28	78.81	76.96	76.46	78.49
	MFB [47]	73.62	73.23	74.42	77.68	76.79	78.55	77.56	76.27	78.03
	MFH [48]	72.95	72.43	74.48	77.3	76.64	78.23	77.42	76.39	77.58
Drink Bottle	**Concat**	75.05	75.12	75.25	74.62	74.91	75.4	75.93	75.15	**77.38**†
	Block [4]	75.18	75.31	75.39	74.17	74.87	74.94	75.91	75.11	76.23
	Mutan [3]	74.48	73.91	74.72	73.62	75.12	76.05	75.95	74.48	75.97
	MLB [21]	74.34	73.02	75.54	73.55	75.42	75.19	76.37	75.07	76.18
	MFB [47]	74.25	74.25	75.21	74.23	74.88	75.84	76.21	74.78	76.01
	MFH [48]	73.99	73.61	75.36	74.77	75.26	75.72	75.98	74.56	75.85

Table 3. Results obtained by employing different fusion strategies on both the Con-Text and Drink Bottle dataset. For presentation purposes acronyms are used to represent each combination of text recognizers (Textspotter (T), E2E_MLT (E), PHOC (P)) and word embeddings (Word2Vec (W), GloVe (G), FastText (F), Fisher Vector (FV)). The † refers to the proposed model.

Text performs better than Word2Vec or GloVe because it can produce embeddings of out of vocabulary words while considering word n-grams which strengthens our conjecture on the importance of morphology of text to solve this task.

4.5. Comparison of Models

Extensive experiments were conducted regarding the different combinations of text recognizers, word embeddings and fusion techniques. Table 3 show the results obtained in both the Con-Text and Drink Bottle dataset.

When introducing fusion techniques to the models, traditional text recognizers such as E2E_MLT performs better in Con-Text compared to Textspotter, thus achieving a higher mAP. The opposite effect is found in the Drink Bottle dataset, in which Textspotter behaves better than its E2E_MLT. It is interesting to note that the PHOCs obtained perform consistently in both datasets, yielding comparable results to the traditional recognizers employed. Regarding the embedding mechanism utilized, morphological embeddings (FastText, PHOC) work better than purely semantic embeddings due to the discriminative space learned.

We can observe that the usage of fusion techniques usually improve the mAP performance obtained on each method aside from the cases when the models employ Fisher Vector features. Nonetheless, in our experiments we have not found a specific fusion technique that can be generalized for every tested method. Each fusion technique increases the performance for a specific model, being MFH and Block slightly more consistent than others. It is necessary to indicate that employing Fisher Vector features obtained from PHOCs consistently achieves the best performance in a general and consistent manner across both datasets.

In order to asses the efficacy of using the Fisher Vector along with another embedding that captures out of vocabulary words while at the same time considering the character morphology, we employ the Fisher Vector obtained from FastText. To this end, FastText employs character n-grams to construct a relevant vectorial representation of a word, thus it also uses syntax of a detected word. The results of the conducted experiments using Fisher Vector features from FastText and PHOC are shown in the last two columns of Tables 3. There are two results to highlight obtained from this experiment. Firstly, working with PHOCs along FVs always yield better performance compared to Fasttext. The cause might be the information captured by Fasttext encapsulates morphology in the form of character n-grams, as well as semantics. Whereas the PHOC is a compact representation based solely on word morphology. Secondly, by combining the explored fusion methods along with Fisher Vectors did not provide a significant advantage. A straightforward concatenation operation between the FV and the visual features reinforces the notion that both modalities contain discriminative and orthogonal features well suited for this task. As an additional advantage, by employing concatenation the model convergences faster while at the same time providing a better performance.

4.6. Qualitative Results

Fine-grained classification probabilities obtained from our model output are depicted in Figure 5. The textual features employed are able to generalize to unseen textual instances or named entities such as the case of bottle brands or business places. We can observe that our model has a hard time reading handwritten text or vertical textual occurrences, thus wrongly predicting a class, such as the example shown at the first row, seventh column. Nonetheless, the model seems to be capturing text morphology, as can be seen on the prediction of the class 'pawn shop'. Finally on the last two samples on each row, there are not enough

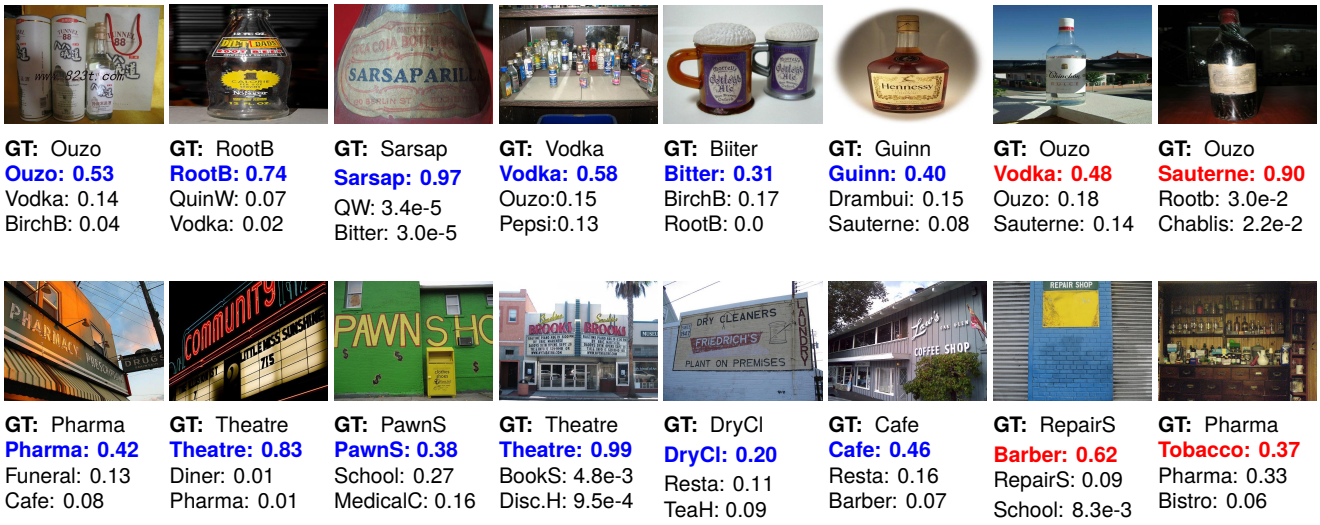

Figure 5. Classification results. The top-3 probabilities of a given image assigned by the output our model is shown along the Ground Truth. Notice that without reading, the classification task is impossible to perform even for humans. Blue and red are used to display correct and incorrect predictions respectively.

guiding textual features and the model relies only on similar visual features. Nonetheless, classifying these samples correctly are a hard task even for humans.

4.7. Fine-grained Image Retrieval

In the same manner as the work presented in [18] and [2], we conduct a retrieval experiment by utilizing the computed vector of the last output layer of the proposed model as retrieval features.

Method	Con-Text	Drink Bottle
Bai*[2]	62.87	60.80
Ours	**64.52**	**62.91**

Table 4. Retrieval results on the evaluated datasets. The results on Con-Text are based on our implementation of the method by [2] since there is no publicly available code. The retrieval scores are depicted in terms of the mAP(%).

We take the approach of query by example, that is, given a sample image that belongs to a specific class, the system must return a ranked list of similar classes as the query. The metric employed to conduct this experiment is the cosine similarity. The proposed method is more robust at the moment of employing a combination of visual and textual features which are discriminative enough to conduct a different task successfully as it is the case in fine-grained image retrieval. The retrieval quantitative performance for both datasets is shown in Table 4, for qualitative results please refer to the Supplementary Material.

5. Conclusions and Future Work

In this work, we have presented a deep neural network framework suitable for a fine-grained classification task. Through extensive experiments conducted, we have presented that leveraging textual information is a key approach to extract information from images. Exploiting these textual cues can pave the road towards more holistic computer vision models of scene understanding. We have shown that current text recognizers that are limited by a dictionary are not the best alternative for this task, because it requires a recognizer able to generalize out of vocabulary words from unseen samples. Additionally, we have analyzed the fact that using semantic embeddings in a fine-grained classification task do not produce the best results due to the related semantic space shared across similar classes. By integrating state-of-the-art techniques and constructing a powerful morphological descriptor from text contained in images, we show that a better suited feature for this task can be learned. Such a feature proves to be useful for a fine-grained classification task as well as for query-by-example image retrieval. Leveraging this robust textual feature yields state-of-the-art results in both tasks across the assessed datasets. Classification and retrieval is possible due to the discriminative features learnt by the model. As future work, we plan to develop a morphological descriptor that captures the same discriminative features using a smaller feature dimension. A continuous valued embedding can replace the binary PHOC while preserving the generalization ability of unseen samples. We want to explore the usefulness of this embedding in other computer vision tasks such as visual question answering [5, 36] and text-based image retrieval.

References

[1] J. Almazán, A. Gordo, A. Fornés, and E. Valveny. Word spotting and recognition with embedded attributes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 36(12):2552–2566, 2014.

[2] X. Bai, M. Yang, P. Lyu, Y. Xu, and J. Luo. Integrating scene text and visual appearance for fine-grained image classification. *IEEE Access*, 6:66322–66335, 2018.

[3] H. Ben-Younes, R. Cadene, M. Cord, and N. Thome. Mutan: Multimodal tucker fusion for visual question answering. In *Proceedings of the IEEE international conference on computer vision*, pages 2612–2620, 2017.

[4] H. Ben-Younes, R. Cadene, N. Thome, and M. Cord. Block: Bilinear superdiagonal fusion for visual question answering and visual relationship detection. *arXiv preprint arXiv:1902.00038*, 2019.

[5] A. Biten, R. Tito, A. Mafla, L. Gomez, M. Rusinol, E. Valveny, C. Jawahar, and D. Karatzas. Scene text visual question answering. October 2019.

[6] P. Bojanowski, E. Grave, A. Joulin, and T. Mikolov. Enriching word vectors with subword information. *Transactions of the Association for Computational Linguistics*, 5:135–146, 2017.

[7] M. Bušta, Y. Patel, and J. Matas. E2e-mlt-an unconstrained end-to-end method for multi-language scene text. In *Asian Conference on Computer Vision*, pages 127–143. Springer, 2018.

[8] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. Imagenet: A large-scale hierarchical image database. In *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*, pages 248–255. Ieee, 2009.

[9] J. Deng, J. Krause, and L. Fei-Fei. Fine-grained crowdsourcing for fine-grained recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 580–587, 2013.

[10] S. Dey, P. Riba, A. Dutta, J. Llados, and Y.-Z. Song. Doodle to search: Practical zero-shot sketch-based image retrieval. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2179–2188, 2019.

[11] L. Gomez, A. Mafla, M. Rusinol, and D. Karatzas. Single shot scene text retrieval. In *The European Conference on Computer Vision (ECCV)*, September 2018.

[12] A. Graves, S. Fernández, F. Gomez, and J. Schmidhuber. Connectionist temporal classification: labelling unsegmented sequence data with recurrent neural networks. In *Proceedings of the 23rd international conference on Machine learning*, pages 369–376. ACM, 2006.

[13] J. Gregor. An algorithm for the decomposition of a distribution into gaussian components. *Biometrics*, pages 79–93, 1969.

[14] A. Gupta, A. Vedaldi, and A. Zisserman. Synthetic data for text localisation in natural images. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2315–2324, 2016.

[15] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[16] T. He, Z. Tian, W. Huang, C. Shen, Y. Qiao, and C. Sun. An end-to-end textspotter with explicit alignment and attention. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5020–5029, 2018.

[17] M. Jaderberg, K. Simonyan, A. Vedaldi, and A. Zisserman. Reading text in the wild with convolutional neural networks. *International Journal of Computer Vision*, 116(1):1–20, 2016.

[18] S. Karaoglu, R. Tao, T. Gevers, and A. W. Smeulders. Words matter: Scene text for image classification and retrieval. *IEEE Transactions on Multimedia*, 19(5):1063–1076, 2017.

[19] S. Karaoglu, J. C. van Gemert, and T. Gevers. Con-text: text detection using background connectivity for fine-grained object classification. In *Proceedings of the 21st ACM international conference on Multimedia*, pages 757–760. ACM, 2013.

[20] D. Karatzas, L. Gomez-Bigorda, A. Nicolaou, S. Ghosh, A. Bagdanov, M. Iwamura, J. Matas, L. Neumann, V. R. Chandrasekhar, S. Lu, et al. ICDAR 2015 competition on robust reading. In *Proc. of the IEEE International Conference on Document Analysis and Recognition*, pages 1156–1160, 2015.

[21] J.-H. Kim, K.-W. On, W. Lim, J. Kim, J.-W. Ha, and B.-T. Zhang. Hadamard product for low-rank bilinear pooling. *arXiv preprint arXiv:1610.04325*, 2016.

[22] K.-H. Kim, S. Hong, B. Roh, Y. Cheon, and M. Park. Pvanet: deep but lightweight neural networks for real-time object detection. *arXiv preprint arXiv:1608.08021*, 2016.

[23] Y. LeCun, Y. Bengio, and G. Hinton. Deep learning. *Nature*, 521(7553):436–444, 2015.

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

[25] M. Liao, B. Shi, X. Bai, X. Wang, and W. Liu. Textboxes: A fast text detector with a single deep neural network. In *AAAI*, pages 4161–4167, 2017.

[26] L. Liu, H. Jiang, P. He, W. Chen, X. Liu, J. Gao, and J. Han. On the variance of the adaptive learning rate and beyond. *arXiv preprint arXiv:1908.03265*, 2019.

[27] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, and A. C. Berg. Ssd: Single shot multibox detector. In *European conference on computer vision*, pages 21–37. Springer, 2016.

[28] L. v. d. Maaten and G. Hinton. Visualizing data using t-sne. *Journal of machine learning research*, 9(Nov):2579–2605, 2008.

[29] S. Maji, E. Rahtu, J. Kannala, M. Blaschko, and A. Vedaldi. Fine-grained visual classification of aircraft. *arXiv preprint arXiv:1306.5151*, 2013.

[30] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean. Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*, pages 3111–3119, 2013.

[31] Y. Movshovitz-Attias, Q. Yu, M. C. Stumpe, V. Shet, S. Arnoud, and L. Yatziv. Ontological supervision for fine

grained classification of street view storefronts. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1693–1702, 2015.

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

[33] F. Perronnin and C. Dance. Fisher kernels on visual vocabularies for image categorization. In *2007 IEEE conference on computer vision and pattern recognition*, pages 1–8. IEEE, 2007.

[34] J. Sánchez, F. Perronnin, T. Mensink, and J. Verbeek. Image classification with the fisher vector: Theory and practice. *International journal of computer vision*, 105(3):222–245, 2013.

[35] B. Shi, X. Bai, and C. Yao. An end-to-end trainable neural network for image-based sequence recognition and its application to scene text recognition. *IEEE transactions on pattern analysis and machine intelligence*, 39(11):2298–2304, 2017.

[36] A. Singh, V. Natarajan, M. Shah, Y. Jiang, X. Chen, D. Batra, D. Parikh, and M. Rohrbach. Towards vqa models that can read. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 8317–8326, 2019.

[37] N. Srivastava and R. R. Salakhutdinov. Multimodal learning with deep boltzmann machines. In *Advances in neural information processing systems*, pages 2222–2230, 2012.

[38] S. Sudholt, N. Gurjar, and G. A. Fink. Learning deep representations for word spotting under weak supervision. *arXiv preprint arXiv:1712.00250*, 2017.

[39] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich. Going deeper with convolutions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1–9, 2015.

[40] P. Tang, X. Wang, B. Feng, and W. Liu. Learning multi-instance deep discriminative patterns for image classification. *IEEE Transactions on Image Processing*, 26(7):3385–3396, 2017.

[41] A. Veit, T. Matera, L. Neumann, J. Matas, and S. Belongie. Coco-text: Dataset and benchmark for text detection and recognition in natural images. *arXiv preprint arXiv:1601.07140*, 2016.

[42] T. Xiao, Y. Xu, K. Yang, J. Zhang, Y. Peng, and Z. Zhang. The application of two-level attention models in deep convolutional neural network for fine-grained image classification. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 842–850, 2015.

[43] H. Xu, G. Qi, J. Li, M. Wang, K. Xu, and H. Gao. Fine-grained image classification by visual-semantic embedding. In *IJCAI*, pages 1043–1049, 2018.

[44] S. Yang, L. Bo, J. Wang, and L. G. Shapiro. Unsupervised template learning for fine-grained object recognition. In *Advances in neural information processing systems*, pages 3122–3130, 2012.

[45] Z. Yang, T. Luo, D. Wang, Z. Hu, J. Gao, and L. Wang. Learning to navigate for fine-grained classification. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 420–435, 2018.

[46] Q. You, H. Jin, Z. Wang, C. Fang, and J. Luo. Image captioning with semantic attention. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4651–4659, 2016.

[47] Z. Yu, J. Yu, J. Fan, and D. Tao. Multi-modal factorized bilinear pooling with co-attention learning for visual question answering. In *Proceedings of the IEEE international conference on computer vision*, pages 1821–1830, 2017.

[48] Z. Yu, J. Yu, C. Xiang, J. Fan, and D. Tao. Beyond bilinear: Generalized multimodal factorized high-order pooling for visual question answering. *IEEE transactions on neural networks and learning systems*, 29(12):5947–5959, 2018.

[49] M. R. Zhang, J. Lucas, G. Hinton, and J. Ba. Lookahead optimizer: k steps forward, 1 step back. *arXiv preprint arXiv:1907.08610*, 2019.

[50] X. Zhou, C. Yao, H. Wen, Y. Wang, S. Zhou, W. He, and J. Liang. East: an efficient and accurate scene text detector. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2642–2651. IEEE, 2017.

[51] Y. Zhu, C. Yao, and X. Bai. Scene text detection and recognition: Recent advances and future trends. *Frontiers of Computer Science*, 10(1):19–36, 2016.