Multi-Modal Embedding for Main Product Detection in Fashion

Antonio Rubio[1, 2] LongLong Yu[2] Edgar Simo-Serra[3] Francesc Moreno-Noguer[1]
[1]Institut de Robòtica i Informàtica Industrial (CSIC-UPC)
[2]Wide Eyes Technologies
[3]Waseda University
arubio@iri.upc.edu, longyu@wide-eyes.it, esimo@aoni.waseda.jp, fmoreno@iri.upc.edu

Abstract

We present an approach to detect the main product in fashion images by exploiting the textual metadata associated with each image. Our approach is based on a Convolutional Neural Network and learns a joint embedding of object proposals and textual metadata to predict the main product in the image. We additionally use several complementary classification and overlap losses in order to improve training stability and performance. Our tests on a large-scale dataset taken from eight e-commerce sites show that our approach outperforms strong baselines and is able to accurately detect the main product in a wide diversity of challenging fashion images.

1. Introduction

Most of current commercial transactions occur online. Every modern shop with growing expectations presents to their potential customers the option of buying some or part of the products in their online catalogs. For instance, 92% of the U.S. Christmas shoppers went online on holidays 2016, a 16.6% more than the same period in 2015 [1].

The way the products are presented to the customer is a key factor to increase online sales. In the case of fashion e-commerce, a specific item being sold is normally depicted worn by a model and tastefully combined with other garments to make it look more attractive. Existing approaches for recommendation or retrieval focus on images only, and normally require hard-to-obtain datasets for training [7], omitting the metadata associated with the e-commerce products such as titles, colors, series of tags, descriptions, etc. that can be used to improve the information obtained from the images.

In this work, we propose to leverage this metadata information to select the most relevant region in an image, or more specifically, to detect the main product in a fashion image that might contain several garments. This allows us to subsequently train specific product classifiers, which do not need to be fed with the whole image. Additionally, this process can also be used as a first step in tasks like visual question answering or, together with customer behavior data, to extract useful information relating the type of images in an e-commerce and its sales.

Figure 1: Overview of our proposed method: from a fashion e-commerce image and its associated textual metadata, we extract several bounding box proposals and select the one that represents the main product being described in the text.

Our approach consists of a first step to extract descriptors of object proposals, that are then used to train joint textual and image embedding. The distances between descriptors in this common latent space are then used to retrieve the main product of each specific image as the closest object proposal to the textual information.

We train our method with images of individual garments and evaluate it in a different dataset of images of models wearing the clothes, and it is able to detect a region with an exigent 70% overlap with the ground truth in more than 80% of the cases among the top-3 bounding box proposals.

2. State of the Art

Our work focuses on the combination of textual and visual information applied to the task of specific object detection (fashion items for our specific case), therefore it lies in

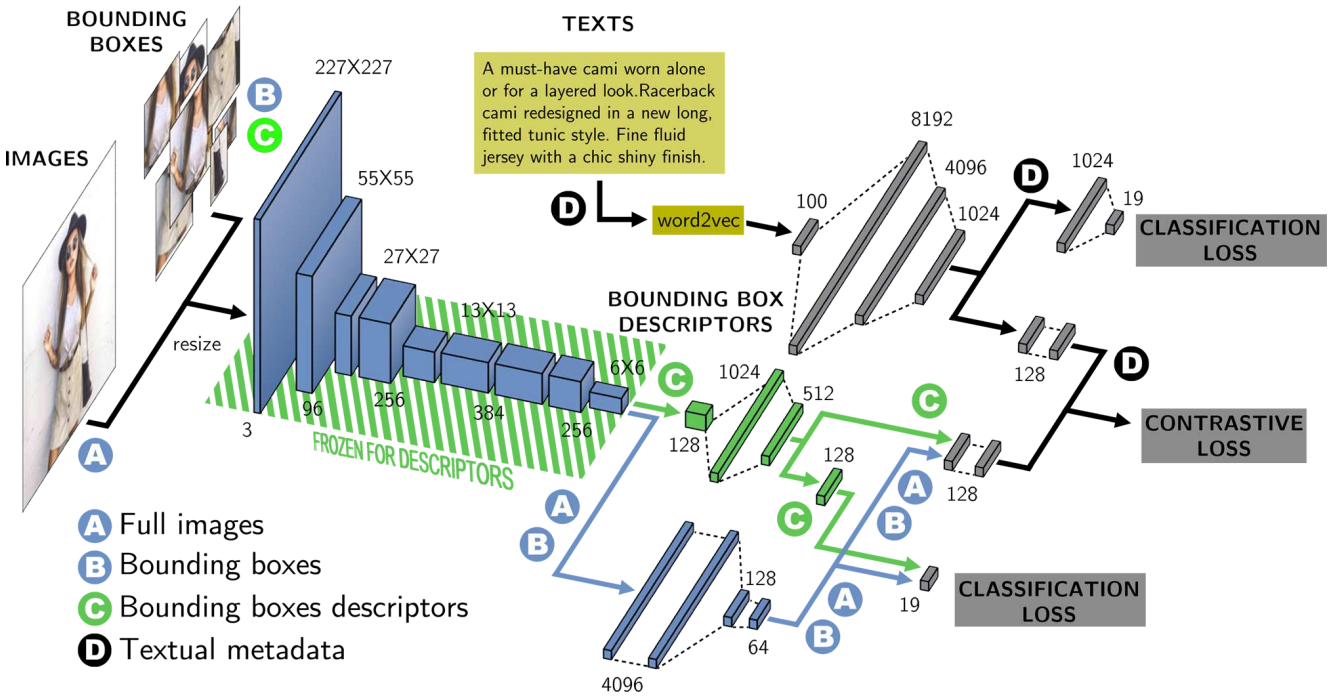

Figure 2: The three different network architectures used in the paper. Gray layers remain constant for all the architectures (i.e., the text branch (D) and a few layers before each loss function). Blue parts correspond to architectures using images as input (both full image (A) and cropped bounding boxes (B) flow through the same layers), and green parts correspond to the architecture using bounding box descriptors as input. These descriptors are the output of the frozen first layers of the AlexNet architecture, so in case (C) the image branch of the network is only trained from the first green layer.

between the fields of multi-modal embedding, object detection and phrase localization.

Fashion is a predominantly visual world, which has led many researchers in the past years to apply Computer Vision techniques to solve specific fashion tasks. Common examples are clothing classification or retrieval [3, 2, 23], clothing parsing (i.e., semantically label each image pixel) [29, 21, 4, 15] or higher level tasks such as evaluating style or deducting people's occupation or social tribe [24, 11, 28, 22, 17].

Many recent works focus on generic (not fashion-specific) multi-modal embeddings for images and text, most of them oriented to automatic multi-labelling of images. DeViSe [5] or ConSE [19], for instance, are text and image embeddings created by using labels from ImageNet and devoted to this task. In MIE [20] the authors use geodesic object proposals [12] to automatically generate multiple labels for images based on meaningful subregions. This approach is similar to our approach in the fact that they find the minimum distance between proposed image regions and texts. Nevertheless, their text data consists of simple ImageNet labels, and they retrieve those labels that are closest to a specific image region, while we find the image region closest to a rich textual description. Furthermore, we base oun approach on a much faster algorithm in [31] to generate the proposals.

Other works try to acquire a deeper understanding of the available textual information. The embedding in [9] is created to explicitly enforce class-analogy preservation. In [26], they deal with the task of image-sentence retrieval using whole images and in the final experiments of the paper they face the phrase localization task on the Flickr30K Entities dataset. They use the same basic idea of two network branches (one for images, one for texts) connected with a margin loss, but we incorporate the classification information to the gradients of the network, while they only enforce the ranking task with combined hinge loss functions. In [14], they propose a two-step process where they first train a network with multi-labeled images and then use this trained network to mine top candidate image regions for the labels. The work of [27] is devoted to the structured matching problem, studying semantic relationships between phrases and relating them to regions of images. Some other works focus on Visual Question Answering [30], taking the goal of image region importance according to text a step further, using it to generate a proper answer to a question. While these works focus on many-to-many correspondences, i.e. relating parts of sentences to regions of images, our work tries to associate all the available textual metadata to only one region of the image, simulating the potential problem we are dealing with: receiving images and text from fashion e-commerces and detecting the product being

sold among all the products on the images.

3. Method

Our goal is to detect the *main product* corresponding to the product being sold in a fashion image. We consider the case where the image contains several other garments and has additional metadata associated to it. We will solve this problem creating a common embedding for images and texts, and then finding the bounding box whose embedded representation is closest to the representation of the text. In order to do so, we explore different architectures and combinations of artificial neural networks. Next, we describe the different approaches that we incrementally propose, stating the pros and cons of each one of them.

3.1. Common parts

Three elements of our method remain unchanged through all the following architectures: the contrastive loss, the classification losses, and the branch that transforms the textual information into an embedded vector.

Contrastive loss: we use the loss function described by Hadsell *et al*. [8] in order to embed the image and text descriptors in the same space so we can compute distances between them. This is the keystone of our method, which makes both types of input data comparable. This loss is expressed as:

$$L_C(v_I, v_T, y) = (1-y)\frac{1}{2}\left(\|v_I - v_T\|_2\right)^2 + (y)\frac{1}{2}\left(\max\left(0, m - \|v_I - v_T\|_2\right)\right)^2 \quad (1)$$

where y is the label indicating whether the two vectors v_I and v_T, corresponding to image and text descriptors respectively, are similar ($y = 0$) or dissimilar ($y = 1$). The value m is the margin value for negative samples. Therefore, both positive and negative pairs image-text must be used in order for the network to learn a good embedding.

Classification loss: the classification loss (for both text and image branches) is just a cross entropy loss comparing the predicted vector (composed of 19 category probabilities) and the ground truth category label (a binary vector of the same size with only one activation) that can be: *vest*, *skirt*, *swimwear*, *suits*, *shorts*, *jumpsuits*, *shoes*, *pants*, *tops*, *hats*, *accessories*, *belts*, *glasses/sunglasses*, *backpack*, *bags*, *outerwear*, *dress*, *sweatshirt/sweaters* or *background*.

In all our trainings, the global loss function L_G is the weighted sum of the contrastive loss (L_C) and the cross entropy loss (L_X) for image and text:

$$L_G = L_C + \alpha L_X(C_I, L_I) + \beta L_X(C_T, L_T) \quad (2)$$

where $C_I(v_i)$ is the output of the image classification branch, L_I is the image label, $C_T(v_T)$ is the output of the text classification branch, L_T is the text label, and α and β are two weighting hyperparameters.

Text network: the textual metadata is used in the same way throughout all the architectures in the paper. We first concatenate all the available string fields (depending on the source of the data, these can be *title*, *description*, *category*, *subcategory*, *gender*, etc.), then we remove numbers and punctuation signs, and compute 100-dimensional *word2vec* descriptors [18] for each word appearing more than 5 times in the training dataset. We compute these descriptors using bi-grams and a context window of 3 words. Finally, we average the descriptors in order to have a single vector representing the metadata of the product. Averaging these distributed representations gave good results as a text descriptor in [25]. The training corpus for the *word2vec* distributed representation consists of over $400,000$ fashion-only textual metadata.

These descriptors are then fed into a 3-layer neural network formed by Fully-Connected (FC) layers with Batch Normalization (BatchNorm) [10] and Rectified Linear Units (ReLU) that finally produce a 1024-dimensional vector, that is later split into two branches as shown in branch **D** of Fig. 2:

- a FC + BatchNorm + ReLU block that reduces the dimension of the vectors to 128 followed by a final FC layer and a SoftMax layer that reduce it to 19 elements corresponding to category probabilities for classification.
- a FC + BatchNorm + ReLU block followed by a FC layer, both with 128-dimensional outputs. The output of the last layer is the descriptor of the text in the common embedding.

3.2. First Approach: Full Image

We firstly train as a baseline the method using the whole images with their associated textual information. These images present a huge variability: for shoes, for instance, they usually consist of a frontal and superior view of one shoe, for pants they show in many cases a model's legs (including feet with shoes) but some shorts appear individually, shirts usually appear also individually, etcetera. We use these images with their associated metadata to train the network. This is the baseline against which we compare. We construct the positive pairs as an image with its corresponding metadata, and the negative pairs as an image with textual metadata of a product from a different category.

The network architecture is shown in Fig. 2. It consists of the previously explained text network joint with the image network in branch **A**, whose architecture adopts the shape of the well-known AlexNet [13] network followed by a few FC + BatchNorm + ReLU layers. Outputs from both branches of the network converge in the contrastive loss. Text and image gradients are also influenced by category classification losses. This approach is a fast, straightforward and not very accurate way of solving this specific

Table 1: Results of the architectures detailed in Section 3, including precision@top-K and classification accuracies.

STRUCTURE	precision@top-K						Classification Accuracy	
	1	3	5	20	50	100	Text	Image
1. Full image	21,87%	44,74%	58,48%	76,06%	79,58%	82,47%	**98.08%**	**90.06%**
2. Bounding boxes	53,52%	70,42%	77,46%	90,07%	92,11%	92,96%	94.22%	88.63%
3. 2 with overlap	52,11%	78,87%	81,69%	**90,24%**	91,30%	91,58%	97.24%	84.74%
4. RoI pooling	**56,34%**	**80,01%**	**84,51%**	90,14%	**92,96%**	**95,77%**	96.91%	80.33%

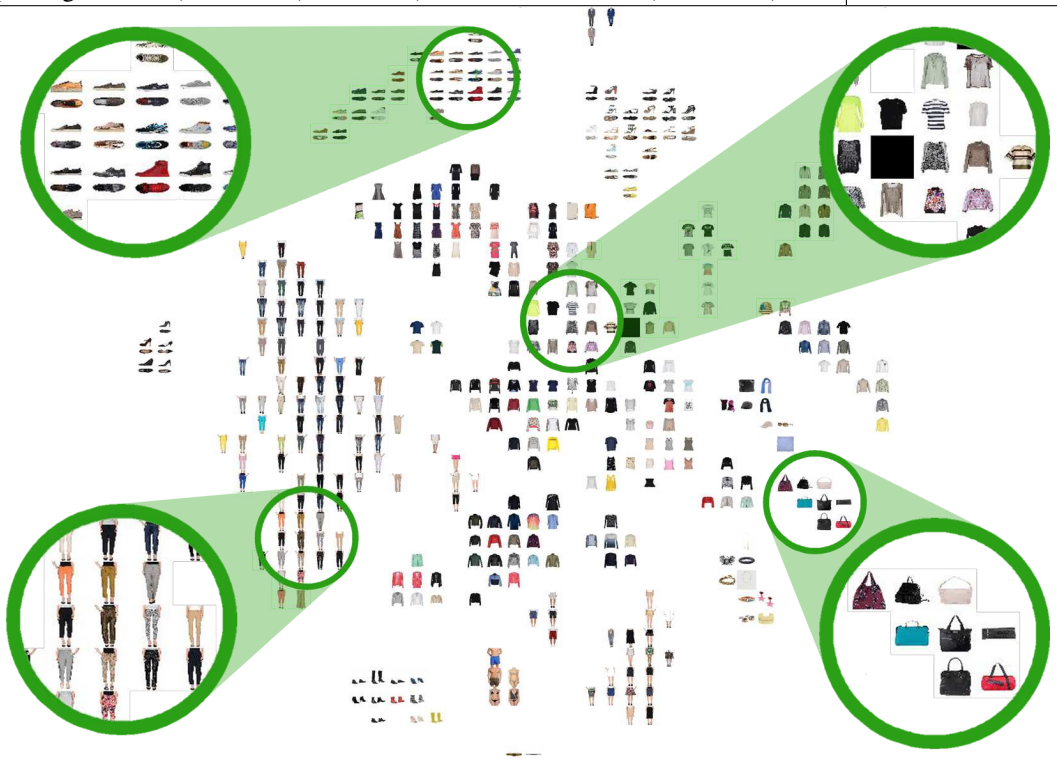

Figure 3: Two-dimensional t-SNE visualization of training set images computed with their projections in our embedding.

problem, and depending on the available data, it might be the only way.

3.3. Second Approach: Bounding Boxes

Since our final goal is to detect the most relevant region of the image according to the text, it makes sense to train the network with smaller parts of the image instead of the whole image itself. In order to do that, we use the Ground Truth bounding boxes (*GT*) of each image, along with 300 bounding boxes per image, computed with [31] (*proposals*) as the input for our network. Now, we define the possible combinations for positive and negative pairs as follows:

- **Positive pairs:** text_i and:
  a) *GT* bounding boxes of image_i
  b) *proposal* bounding boxes of image_i with overlap of over 70% with *GT*.
- **Negative pairs:** text_i and:
  a) *proposal* bounding boxes of image_i whose overlap with *GT* is between 30% and 50%.
  b) *proposal* bounding boxes of image_j (from a different category) with overlap of over 70% with their *GT*

For this approach, the network is the same as before (see branch **B** of Fig. 2), but the input pairs are the resized *proposal* bounding boxes with their corresponding positive or negative texts. The quality of the results is considerably increased, but since the number of pairs that we can construct per product is now much higher, it will take more time to reach a good minima when training.

3.4. Third Approach: Region of Interest Pooling

After shifting from whole images to bounding boxes, the number of positive and negative pairs that can be fed into the network is highly increased. Therefore, the training process takes more time. For this reason, our next step is to train a smaller network with compact representations of these images, reducing the computational cost of training all the convolutional layers of the previous architecture. Positive and negative pairs are constructed in the same way, but in this case the input to the image part of the network are

the $6 \times 6 \times 256$ Region of Interest (RoI) pooling regions of the corresponding *proposal* bounding boxes extracted from the last convolutional layer of AlexNet as in [6].

Now the training of the visual part of the network consists only of a first convolutional layer (coupled with a ReLU) that reduces the third dimension of the data from 256 to 128 elements, followed by two FC + BatchNorm + ReLU blocks that progressively transform these descriptors into 512-dimensional vectors. Layers previous to this first convolutional layer are frozen and only used to extract the RoI pooling features (see branch **C** of Fig. 2).

3.5. Overlap Loss

Since our goal is to maximize the overlap between the selected bounding box (the one whose descriptor is closest to the text descriptor) and the ground truth bounding box, we try to add this overlap information to the embedded descriptors. We do it learning to predict the overlap of each bounding box with the corresponding ground truth of its image using a L1 regression loss.

In this case, the full training loss L_{GO} consists of the previous loss summed with the weighted overlap loss:

$$L_{GO} = L_G + \gamma L_O(\hat{ov}, ov) \tag{3}$$

where L_O is the L1 regression loss for overlap, $\hat{ov}$ is the predicted overlap of the bounding box with the corresponding ground truth bounding box and ov is the actual overlap with the ground truth, computed as their intersection. This case is omitted from Fig. 2 for clarity.

All the design choices for the network layers were taken after meticulous ablation studies.

4. Dataset

Our training and validation dataset consists of $458,700$ products from eight different e-commerces. Our testing dataset consists of $3,000$ products coming from a different e-commerce, and will be made publicly available[1]. Each product from the dataset is formed by an image with the annotated *GT* bounding box and its associated metadata. Some examples of the images and their associated textual information can be seen in Fig. 4.

5. Results

The performance of our method was evaluated using a dataset different from the training dataset in order to test its ability to generalize. In all the cases, the networks were trained with batches of 64 pairs, with $\alpha = \beta\ [= \gamma] = 1$, using stochastic gradient descent with an initial learning rate of 10^{-3} that decreases every $10,000$ iterations by $5 \cdot 10^{-4}$ with momentum 0.95. The margin for the contrastive loss was set to 1 after several tests with different values. During training, classical data augmentation techniques were applied to the images (random horizontal flip, small rotations, etc.). For the bounding boxes, we added random noise to their size and position of up to 5% of the bounding box dimensions. Also, instead of directly resizing every bounding box to the size required by the network ($227 \times 227 \times 3$), they were padded to be as square as the original image dimensions allow to prior to the resizing step, thus taking into account image context and avoiding heavy deformations.

All the results shown in this section come from the following evaluation procedure:

1. Extract the descriptors of the text and the image proposals.
2. Compute the distance between the image and text descriptors, and select the bounding box with the smallest distance to the text.
3. Check the overlap between this bounding box and the ground truth bounding box of the correct product. If the overlap is greater than 70%, the result is considered as a positive main product prediction for this image. Otherwise, as negative. Overlap between bounding boxes A and B is computed as $(A \cap B) \,/\, (A \cup B)$.

The numerical results we give in this section are the percentage of test images with positive predictions (overlap with ground truth greater than 70%) from the test set. Evaluations were carried out for different positive overlap percentages, but we consider 70% as a good value. The tendencies for the rest of overlap percentages evaluated were similar.

The first dataset is homogeneously distributed into train and validation pairs of images and metadata (70% for training, 30% for validation). For each network architecture, we use for testing the weights values of the iteration with best performance in the validation subset. The test dataset (from which we present results) comes from a different image source to prove the generalization ability of the method.

5.1. Quantitative results

In Table 1 we show results of the four architectures explained in Section 3. As expected, every architecture using bounding boxes surpasses the basic architecture using the whole image in terms of percentage of test images with any of the top-K retrieved bounding boxes overlapping more than 70% with the ground truth. We see that incorporating the overlap information increases the performance of the method. Also, in general, the approach using RoI pooling descriptors yields better results than the approach using bounding boxes through the whole image architecture. For the architecture predicting the overlap percentage between each *proposal* and the *GT* bounding box, the average error in the percentage prediction is $5,81\%$. Even though our

[1]Test dataset including images, main product bounding boxes and textual metadata will be public on the author's website.

(a) **Category:** Coats & Jackets. **Description:** Women >Coats Canada Goose - Trillium parka. Canada Goose at Simons. Designed to withstand extreme conditions, the Trillium parka keeps you snug and warm, even in the depths of winter. Specially styled for women, it features a sleek fit, slightly cinched waist, and slimmer lines throughout. Dual-adjusting removable hood with removable coyote fur ruff Interior shoulder straps to carry the parka like a backpack Heavy-duty locking zipper with snap-button storm flap. Upper fleece-lined pockets, lower flap pockets with snaps. Made in Canada. The model is wearing size small. **Title:** Canada Goose - Trillium parka (Women, Pink, XX-SMALL). **Color:** Khaki.

(b) **Category:** Coats & Jackets. **Description:** Women >Coats. Vero Moda - Long baseball jacket Vero Moda. Vero Moda at Icône A preppy, chic and sporty piece for a trendy fall look. Blended wool with an ultra soft brushed finish and fine satiny lining. Ribbed knit collar and cuffs Snap closure. Zip pockets. The model is wearing size small. **Title:** Vero Moda - Long baseball jacket (Women, Black, X-SMALL). **Color:** Black.

(c) **Category:** Skirts. **Description:** Women >Skirts. Contemporaine - Straight belted skirt Contemporaine. Exclusively from Contemporaine. Versatile straight fit accented by chic, decorative metallic zippers. Thin faux-leather belt included. Invisible zip closure behind. Structured stretch cotton-blend weave with a flawless fit, fine and silky built-in full-length lining. Matching items also available. The model is wearing size 4. **Title:** Contemporaine - Straight belted skirt (Women, Blue, 8). **Color:** Black.

(d) **Category:** Sweaters & Cardigans **Description:** Women >Sweaters & Cardigans Icone - Zipped polo-collar sweater Icone Exclusively from Icone. An essential updated with a contrasting collar and ring zipper for a graphic techno look. Fine stretch knit in a cotton-modal blend. Ribbed edging. The model is wearing size small. **Title:** Icone - Zipped polo-collar sweater (Women, White, X-SMALL). **Color:** White.

(e) **Category:** T-Shirts. **Description:** Women >T-Shirts. Twik - Boyfriend tee Twik. Exclusively from Twik. An ultra practical must-have neutral basic. Ultra comfortable 100% cotton weave. Sewn rolled sleeves. The model is wearing size small. **Title:** Twik - Boyfriend tee (Women, Green, X-SMALL). **Color:** Mossy Green.

Figure 4: Some results of our method. Ground truth is shown in green, and the proposal closest to the text in blue. On top of each figure there is its category and the overlap percentage between the result and the *GT*. Caption of each figure is its textual metadata.

main purpose is not classification, we use it as a help to incorporate to our embedded descriptors the ability to separate better the clothes from different categories. Percentages of classification accuracy for the different architectures are shown in Table 1.

5.2. Qualitative Results

A two-dimensional t-SNE [16] visualization of our embedding is shown in Fig. 3. The images depicted in the embedding come from the training set, and we can see how our method, helped by the category classification losses, learns how to group these images into category clusters. Training set images normally present an individual garment over a white background.

Then, results of using this embedding to perform the task of main product detection in the test set can be seen in Fig. 4. There, some images with their associated metadata and *GT* bounding box are shown along with the nearest *proposal* detected by our method. Note how these pictures are different from the training set: they normally show the products worn by models, who also wear other clothes that might partially or completely appear on the images. Sometimes the background is textured (Fig. 4 (a) (d)).

6. Conclusions

We present a method that uses textual metadata to detect the interest product in fashion e-commerce images. We represent the text using a distributed representation and for our best approach we use compact representations of bounding boxes extracted from frozen layers of a pre-trained network. We compare several network architectures combining different loss types (contrastive, cross-entropy and L1 regression). In our test dataset, with images and texts coming from a different e-commerce than those used for training, our method is able to rank the main product bounding box in the top-3 most probable candidate bounding boxes among 300 candidates in an 80% of the cases. At the same time, the network learns to classify these products into the corresponding clothing category with high accuracy.

Acknowledgments: This work is partly funded by the Spanish MINECO project RobInstruct TIN2014-58178-R. A.Rubio is supported by the industrial doctorate grant 2015-DI-010 of the AGAUR. The authors are grateful to the NVIDIA donation program for its support with GPU cards.

References

[1] The Ultimate List of E-Commerce Stats for Holiday 2016. https://goo.gl/MBRUJo. Accessed: 2017-01-23. 1

[2] L. Bossard, M. Dantone, C. Leistner, C. Wengert, T. Quack, and L. Van Gool. Apparel classification with style. In *ACCV*, 2012. 2

[3] W. Di, C. Wah, A. Bhardwaj, R. Piramuthu, and N. Sundaresan. Style finder: Fine-grained clothing style detection and retrieval. In *CVPR Workshops*, 2013. 2

[4] J. Dong, Q. Chen, W. Xia, Z. Huang, and S. Yan. A deformable mixture parsing model with parselets. In *CVPR*, 2013. 2

[5] A. Frome, G. S. Corrado, J. Shlens, S. Bengio, J. Dean, T. Mikolov, et al. Devise: A deep visual-semantic embedding model. In *NIPS*, 2013. 2

[6] R. Girshick. Fast R-CNN. In *Proceedings of the IEEE international conference on computer vision*, pages 1440–1448, 2015. 5

[7] M. Hadi Kiapour, X. Han, S. Lazebnik, A. C. Berg, and T. L. Berg. Where to buy it: Matching street clothing photos in online shops. In *CVPR*, 2015. 1

[8] R. Hadsell, S. Chopra, and Y. LeCun. Dimensionality reduction by learning an invariant mapping. In *CVPR*, 2006. 3

[9] S. J. Hwang, K. Grauman, and F. Sha. Analogy-preserving semantic embedding for visual object categorization. In *Proceedings of the 30th International Conference on Machine Learning (ICML-13)*, pages 639–647, 2013. 2

[10] S. Ioffe and C. Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *ICML*, 2015. 3

[11] M. H. Kiapour, K. Yamaguchi, A. C. Berg, and T. L. Berg. Hipster wars: Discovering elements of fashion styles. In *ECCV*, 2014. 2

[12] P. Krähenbühl and V. Koltun. Geodesic object proposals. In *European Conference on Computer Vision*, pages 725–739. Springer, 2014. 2

[13] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In *NIPS*, 2012. 3

[14] D. Li, H.-Y. Lee, J.-B. Huang, S. Wang, and M.-H. Yang. Learning structured semantic embeddings for visual recognition. *arXiv preprint arXiv:1706.01237*, 2017. 2

[15] S. Liu, J. Feng, C. Domokos, H. Xu, J. Huang, Z. Hu, and S. Yan. Fashion parsing with weak color-category labels. *IEEE Transactions on Multimedia*, 16(1):253–265, 2014. 2

[16] L. v. d. Maaten and G. Hinton. Visualizing data using t-sne. *Journal of Machine Learning Research*, 9(Nov):2579–2605, 2008. 6

[17] K. Matzen, K. Bala, and N. Snavely. Streetstyle: Exploring world-wide clothing styles from millions of photos. *arXiv preprint arXiv:1706.01869*, 2017. 2

[18] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean. Distributed representations of words and phrases and their compositionality. In *NIPS*, 2013. 3

[19] M. Norouzi, T. Mikolov, S. Bengio, Y. Singer, J. Shlens, A. Frome, G. S. Corrado, and J. Dean. Zero-shot learning by convex combination of semantic embeddings. *arXiv preprint arXiv:1312.5650*, 2013. 2

[20] Z. Ren, H. Jin, Z. Lin, C. Fang, and A. Yuille. Multi-instance visual-semantic embedding. *arXiv preprint arXiv:1512.06963*, 2015. 2

[21] E. Simo-Serra, S. Fidler, F. Moreno-Noguer, and R. Urtasun. A High Performance CRF Model for Clothes Parsing. In *Proceedings of the Asian Conference on Computer Vision (ACCV)*, 2014. 2

[22] E. Simo-Serra, S. Fidler, F. Moreno-Noguer, and R. Urtasun. Neuroaesthetics in fashion: Modeling the perception of fashionability. In *CVPR*, 2015. 2

[23] E. Simo-Serra and H. Ishikawa. Fashion Style in 128 Floats: Joint Ranking and Classification using Weak Data for Feature Extraction. In *Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016. 2

[24] Z. Song, M. Wang, X.-s. Hua, and S. Yan. Predicting occupation via human clothing and contexts. In *CVPR*, 2011. 2

[25] V. Vukotić, C. Raymond, and G. Gravier. Multimodal and crossmodal representation learning from textual and visual features with bidirectional deep neural networks for video hyperlinking. In *Proceedings of the ACM workshop on Vision and Language Integration Meets Multimedia Fusion*, 2016. 3

[26] L. Wang, Y. Li, and S. Lazebnik. Learning deep structure-preserving image-text embeddings. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5005–5013, 2016. 2

[27] M. Wang, M. Azab, N. Kojima, R. Mihalcea, and J. Deng. Structured matching for phrase localization. In *European Conference on Computer Vision*, pages 696–711. Springer, 2016. 2

[28] K. Yamaguchi, T. L. Berg, and L. E. Ortiz. Chic or social: Visual popularity analysis in online fashion networks. In *ACMMM*, 2014. 2

[29] K. Yamaguchi, M. Hadi Kiapour, and T. L. Berg. Paper doll parsing: Retrieving similar styles to parse clothing items. In *CVPR*, 2013. 2

[30] B. Zhou, Y. Tian, S. Sukhbaatar, A. Szlam, and R. Fergus. Simple baseline for visual question answering. *arXiv preprint arXiv:1512.02167*, 2015. 2

[31] C. L. Zitnick and P. Dollár. Edge boxes: Locating object proposals from edges. In *ECCV (5)*, pages 391–405, 2014. 2, 4