To appear in AMC MM'11. Note that this paper has three additional authors (Ping Wang, Atul Kanaujia, and Niels Haering) as listed in the acknowledgements as suggested by the publication guidelines.

Fusing object detection and region appearance for image-text alignment [*]

Luca Del Pero Philip Lee James Magahern Emily Hartley Kobus Barnard
University of Arizona
{delpero, phlee, jamesmag, elh, kobus}@email.arizona.edu

ABSTRACT

We present a method for automatically aligning words to image regions that integrates specific object classifiers (e.g., "car" detectors) with weak models based on appearance features. Previous strategies have largely focused on the latter, and thus have not exploited progress on object category recognition. Hence, we augment region labeling with object detection, which simplifies the problem by reliably identifying a subset of the labels, and thereby reducing correspondence ambiguity overall. Comprehensive testing on the SAIAPR TC dataset shows that principled integration of object detection improves the region labeling task.

Categories and Subject Descriptors

I.4.8 [**Image processing and computer vision**]: Scene analysis—*Object recognition*

General Terms

Algorithms, Performance

1. INTRODUCTION

There has been much recent work on both training classifiers for object recognition, as well as learning models for automated image annotation, from weakly labeled data (e.g., [2, 5, 7, 13, 17, 16]). Such data consists of pictures with associated keywords, where it is not known which word corresponds to which part of the image (correspondence ambiguity). Learning from weakly labeled images is challenging, since it requires solving two problems at once: 1) locating the desired entity in the training data; 2) learning a model for appearance. Recent work suggests addressing the correspondence ambiguity in weakly labeled data as a separate problem from learning object appearance [3, 11]. Specifically, the goal is to align each label, initially associated with the whole picture, with the correct part of the image (Figure 1). We refer to this task as aligning the training data. Once this is done, this augmented data can be used for learning region or object models. Further, we agree with others that labels have to be localized in order to be useful for retrieval and recognition [1, 3].

In this work, we contribute a new strategy for aligning the training data. Specifically, we propose to combine object detectors, designed for robustly identifying a single specific entity, and existing methods for linking words to regions based on weak models. Previous work (e.g., [3, 11]) has relied on the latter, since it is not known in advance which models are good for which objects or regions. For example, some are better defined in terms of shape (e.g. "cup"), some in terms of color or texture ("zebra"), and some might require a part-based model ("bicycle"). However, developing a specific model for every word is impractical and does not scale. On the other hand, simple generic models can generalize over a large number of different entities, but cannot achieve the fine level of tuning required for robustly identifying a specific entity. It then seems natural to exploit existing detectors to reliably align some of the labels, relying on weak models for the remaining ones.

We also argue that detectors are useful for annotation when combined with exclusion reasoning [3], which posits that each word associated to an image should be assigned to at least one of its regions. Consider the example in Figure 1. If a car detector identifies the central region as "car", the labeling task becomes easier, as the choice for the two remaining regions is now restricted to "sky" and "snow". This also accommodates for the fact that detectors are usually available for common objects only. In fact, less common objects can be aligned via exclusion reasoning, thus producing data for training robust detectors for them.

The main contribution of this work is a principled strategy for integrating object detectors and exclusion reasoning into existing algorithms for region labeling. Specifically, we propose a probabilistic framework, as this is an effective way to combine different sources of information in this domain [6, 12, 15]. To test our ideas, we incorporate state-of-the art detectors [10] into a recent algorithm for region labeling [3].

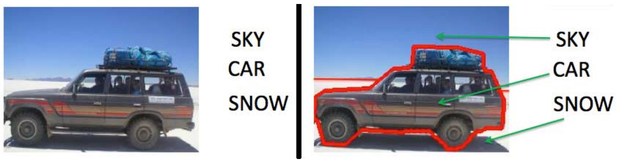

Figure 1: Solving the correspondence ambiguity in weakly labeled data. See text for details

[*] Area chair: Alexander Hauptmann

MM'11, November 28–December 1, 2011, Scottsdale, Arizona, USA.
Copyright 2011 ACM 978-1-4503-0616-4/11/11 ...$10.00.

2. OVERVIEW OF OUR APPROACH

We start by defining a probabilistic framework for labeling the training data at the region level. Our input is a set of images, each with a set of associated keywords. We define the vocabulary V as the union of all such keywords $(w_1, ..., w_{|V|})$. Each image is subdivided into regions using a standard segmentation algorithm based on low level features, such as color and texture (for this work, we used NCuts [14]). We use the notation r_j to identify a generic image region produced by the segmenter.

Our goal is to compute a distribution $p(w|r_j)$ for each region over the whole vocabulary. The probabilistic constraints

$$\sum_{w \in V} p(w|r_j) = 1 \tag{1}$$

embody an additional manifestation of exclusion reasoning, which states that to the extent that we believe r_j is to be labeled with a word, it should not be labeled with other words. In fact, a word w_k with high $p(w_k|r_j)$ results in small $p(w_u|r_j)$ $\forall u \neq k$. For example, if a car detector outputs a low (high) probability $p(car|r_j)$ for region r_j, this will result in a high (low) probability $p(w_k|r_j)\forall w_k \neq$ "car". Note that this allows to smoothly integrate detectors, which are binary classifiers, into our region labeling scenario, which is a multiclass labeling problem. However, this requires mapping the output of detectors (typically scores) into probabilities using held out data, as discussed in Section 3.

Given $p(w|r_j, X)$ and $p(w|r_j, Y)$ computed using different sources of evidence X (e.g. detectors) and Y (e.g. appearance), we can combine them into $p(w|r_j, X, Y)$ by simple multiplication or by averaging [15]. When all information is incorporated into an output posterior distribution $p(w|r_j,$ All constraints), we simply label r_j with the word w_k maximizing this distribution. In Section 4, we will discuss a strategy to integrate the calibrated output of a set D of detectors $p(w_k|r_j, D)$ with a posterior distribution $p(w_k|r_j)$, which can be computed using any other source of evidence. In order to test this approach, we will first use a state-of-the-art algorithm [3] that uses exclusion reasoning and image appearance to compute a first estimate of $p(w_k|r_j)$ $\forall j$, which we call $p(w_k|r_j, baseline)$. This in fact the baseline we want to augment with $p(w_k|r_j, D)$, producing $p(w_k|r_j, baseline, D)$.

The baseline labeling algorithm. In this section, we summarize the labeling approach by Barnard and Fan [3], that we will extend with object detectors. Given the problem formulated above, this algorithm estimates the unknown posterior distributions $p(w|r_j)$ using four main constraints: 1) similar regions from different images are likely to be labeled with the same word, if they come from images sharing at least one keyword; 2) the label for a region should be chosen among the keywords associated with the image containing it; 3) each image label should have at least one region associated with it (exclusion reasoning); and 4) $\forall i$, $p(w|r_i)$ is a distribution over V, implying $\sum_{(w_v \in V)} p(w_v|r_i) = 1$; and $0 \leq p(w_v|r_i) \leq 1 \forall v, i$. See Barnard and Fan[3] for more detail. In what follows we will refer to the output of this algorithm as $p(w|r_j, baseline)$. We now describe how we calibrate object detectors output so that it can be integrated with this distribution.

3. CALIBRATING DETECTOR OUTPUT

Existing object detectors typically attach detection scores to specific portions of images, such as bounding boxes or super-pixels (Figure 2, top row). Classification is then done by comparing the score for an image region against a threshold. We need to convert such scores into probabilities before they can be used in our probabilistic framework. Specifically, we need

$$p(w_k|r_j, d_k) \quad , \tag{2}$$

which is the probability of assigning word w_k to image region r_j given the information provided by detector d_k (trained to recognize word w_k) for that particular region.

We develop a calibration procedure to map detection scores to probabilities based on training images where objects of interest have been manually identified. This is very different than simply scaling the score to the interval [0,1], and allows further reasoning when combined with probabilities coming from different sources, which the raw score alone would not.

We define s_{kj} to be the score assigned to r_j by a detector d_k. Our goal is to estimate $p(w_k|r_j, d_k) = p(w_k|s_{kj})$. Intuitively, we approximate these probabilities with the number of correct identifications at the given score (ie, the number of correct identifications at score s), divided by the number of times the detector output score s, at a suitable discretization. First, we define

$$K_m = \{r_1, ..., r_{|K_m|}\}, K_s = \{r_1, ..., r_{|K_s|}\} \quad . \tag{3}$$

The former is the set of image regions in the training data that were manually labeled with word w_k, while the latter is the set of image regions in the training data for which the detector d_k output score s. We estimate $p(w_k|s)$ as the number of pixels for which d_k output score s AND that were also manually labeled with w_k, divided by the total number of pixels for which d_k output score s

$$p(w_k|s) = \frac{\sum\limits_{r_i \in K_m, r_j \in K_s} size(r_i \cap r_j)}{\sum\limits_{r_j \in K_s} size(r_j)} \quad . \tag{4}$$

Here, the operator $size()$ measures the size of a region in terms of pixels, and $r_i \cap r_j$ is the overlap between r_i and r_j. The score is discretized using soft binning.

Given a score s_{kj} for a superpixel or bounding box r_j computed by detector d_k, we are now able to compute

$$p(w_k|r_j, d_k) = p(w_k|s_{kj}) \tag{5}$$

using the distribution calibrated with the procedure just described. However, we are usually interested in computing $p(w_k|r_i)$ for a region r_i produced by a segmentation algorithm (Figure 2, middle left), which does not directly correspond to detector output. Typically, given an image I, a detector d_k provides a set of possibly overlapping regions $K_I = (r_1, .., r_N)$ with corresponding scores $SK_I = (s_1, ..., s_N)$. We convert these scores into probabilities $PK_I = (p_1, ..., p_N)$. Last, the probability $p(w_k|r_i, d_k)$ for any region r_i in I can be computed either as the mean probability over the region

$$p(w_k|r_i, d_k) = \frac{\sum\limits_{r_u \in K_I} (p_u * size(r_u \cap r_i))}{size(r_i)} \quad , \tag{6}$$

or as the max

$$p(w_k|r_i, d_k) = \max_{(r_u \in K_I : size(r_u \cap r_i) > 0)} (p_u) \quad . \tag{7}$$

Calibration at multiple scales. Several object detectors achieve scale invariance by providing scores at multiple scales. The procedure discussed in the previous section can be extended to calibrate $p_\sigma(w_k|s^\sigma)$ at a particular scale σ, provided that the size of the manual regions in K_I^σ used for calibration roughly match σ (s^σ is the score at scale σ). Given a detector d_k operating at multiple scales $\Sigma = (\sigma_1, ..., \sigma_{N_\sigma})$, we estimate $p_\sigma(w_k|s^\sigma)\forall\sigma \in \Sigma$. Then, we modify Equation 6 and 7 to

$$p(w_k|r_j, d_k) = \max_{\sigma\in\Sigma} \frac{\sum_{r_u^\sigma \in K_I^\sigma} (p_u^\sigma * size(r_u^\sigma \cap r_j))}{size(r_j)} \quad , \qquad (8)$$

and

$$p(w_k|r_j, d_k) = \max_{\sigma\in\Sigma} \max_{(r_u^\sigma \in K_I^\sigma : size(r_u^\sigma \cap r_j) > 0)} (p_u^\sigma) \quad . \qquad (9)$$

We use the max in both cases, since detectors perform classification by comparing the maximum over scale space against a threshold.

4. INTEGRATING OBJECT DETECTORS

In this section we propose a strategy for combining the posterior distribution $p(w|r_j, baseline)$ with the contributions of specific detectors, which we can now convert into probabilities. Combining $p(w|r_j, baseline)$ over the whole vocabulary, and probability $p(w_k|r_j, d_k)$ computed using detector d_k is not straightforward. While the former is a probability distribution over the vocabulary, and thus sums to one, $p(w_k|r_j, d_k)$ can be interpreted as the output of a binary classifier for word w_k, and does not allow to make any assumptions on the value $p(w_n|r_j, d_k)$ when $n \neq k$. However, we can assume that $(1-p(w_k|r_{ij}, d_k))$ is the probability that the region is to be labeled with a word that is not w_k given the evidence provided by detector d_k. This suggests that we could weight $p(w_n|r_j, baseline)$ with $p(w_k|r_j, d_k)$ when $n = k$, and with $(1 - p(w_k|r_{ij}, d_k))$ otherwise

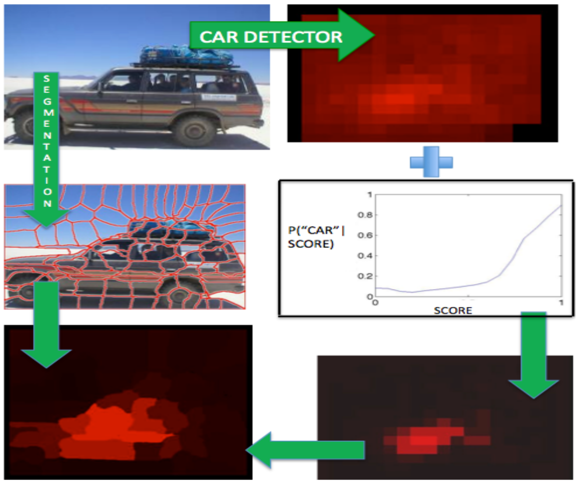

Figure 2: Mapping score into probabilities. Here, a car detector provided a score map (top right) for the original image, visualized by setting the red channel of each cell proportionally to the score (min score r = 0, max score r = 255). Using our calibration procedure, we found a mapping $p(''car''|score)$ (middle right), that we use to convert the score map into a probability map (bottom right), visualized by setting the red channel proportionally to the probability value (p(”car”|s)=0 implies r = 0, p(”car”|s)=1 implies r = 255). This is then used to compute probabilities for the image regions of interest (bottom left), for example the output of a segmentation algorithm (middle left).

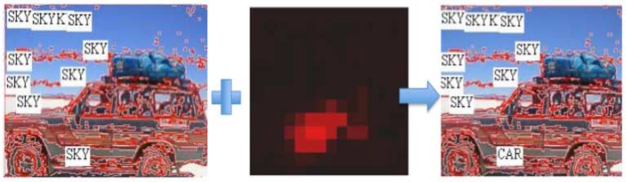

Figure 3: Benefits of incorporating a car detector in the labeling process, see text for details

$$p(w_n|r_j, d_k, baseline) \propto p(w_n|r_j, baseline) * \\ p(w_k|r_j, d_k)^{\delta(n,k)}(1 - p(w_k|r_j, d_k))^{1-\delta(n,k)} \quad , \qquad (10)$$

where $\delta(n, k) = 1$ if $n = k$, 0 if they are different. An example is shown in Figure 3. Here, we see the labels produces by the baseline algorithm (left), and the labels maximizing Equation 10 (right) after adding a car detector (middle).

Suppose now we have a set $D = d_1, ..., d_{|D|}$ instead of a single one. We modify Equation 10 as

$$p(w_n|r_j, D, baseline) \propto p(w_n|r_j, baseline) * \\ \prod_{d_k \in D} (p(w_k|r_j, d_k)^{\delta(n,k)}(1 - p(w_k|r_j, d_k))^{1-\delta(n,k)}) \quad . \qquad (11)$$

Importantly, this formulation is generic enough to augment any approach capable of computing a posterior distribution $p(w|r)$ over the vocabulary.

As described, we use a specific detector d_k on a region j only if w_k is in the label for the image containing r_j. However, we extend this by including words that are related to the detector that are likely to be visually similar. Specifically, we use a detector for a given word if any of its synonyms and hyponyms are in the image label set. For example, we use the detector for word w_k (eg, “car”) on both its synonyms (”automobile”) and its hyponyms (”Station Waggon”, “Compact”). We use the WordNet semantic hierarchy [9] to get the hyponyms and synonyms.

5. RESULTS AND CONCLUSIONS

All our tests were performed on the SAIAPR TC-12 Benchmark [8], consisting of 20000 images, each annotated with 2 to 10 keywords. Ground truth manual segmentations and annotations are available for each image. We use a comprehensive evaluation strategy [4] for comparing the labels produced by our algorithm against ground truth annotations at the region level. In this scope, we used two different measures: 1) the frequency of correct labels predicted by an algorithm (“frequency”), and 2) the number of words an algorithm can reliably identify (“range”) [4].

We tested our full approach on seven state-of-the-art detectors [10], listed in Table 2, for which we had enough images for both calibration and testing. First, each detector was calibrated on 80 positive and 40 negative images. We then tested on seven splits of 200 images, where each image was labeled with at least one of the seven words we have a detector for. Each image was segmented using NCuts [14].

Table 1: Region word alignment performance

Algorithm	Range	Frequency
Empirical distribution	0.14	2.75
Baseline	1.08	4.60
Detectors (avg)	1.24	5.59
Detectors (max)	1.28	6.04
Theoretical maximum	4.41	19.96

Table 2: Results where each detector is relevant

Object	Range	Freq	Object	Range	Freq
Bicycle	+58.8%	+36.3%	**Car**	+22.4%	+46.3%
Bird	+3.0%	+0.1%	**Chair**	+17.8%	+39.5%
Boat	+8.4%	+33.7%	**Person**	+18.9%	+19.9%
Bottle	+10.2%	+44.9%			

The results in Table 1 show that adding detectors improves performance of the baseline, both in terms of range and frequency. We tested two different ways of computing $p(w_k|r_j, d_k)$ (Equation 8 and 9), and we found slightly better results when using the maximum over a region. A few qualitative results are shown in Figure 4.

Table 2 shows results restricted to images where a given detector was used, for each detector, using Equation 9 to combine results. Here we tabulate the improvement over the baseline, which is generally substantive. This validates that when we have an appropriate detector, using it generally improves alignment.

Conclusions. We developed a general approach to integrate object specific detectors with region labeling based on weak models for appearance. We advocate two contributions: 1) A generic formulation for converting the output of detectors into probabilities; 2) a way to integrate these probabilities, coming from binary classifiers for specific words, with posterior distributions over the whole vocabulary.

We plan to extend our strategy to incorporate additional sources of information. One example is where labels have associated adjectives such "red" or "white", which can provide prior probabilities on the label, and could be exploited similarly to the way we use detectors. A second source of information that we would like to exploit is prior knowledge from spatial context.

6. ADDITIONAL AUTHORS

Ping Wang, Atul Kanaujia, Niels Haering, ObjectVideo, {pwang, akanaujia, niels}@objectvideo.com

7. REFERENCES

[1] L. H. Armitage and P. G. B. Enser. Analysis of user need in image archives. *Journal of Information Science*, 23(4):287–299, 1997.

[2] K. Barnard, P. Duygulu, N. d. Freitas, D. Forsyth, D. Blei, and M. I. Jordan. Matching words and pictures. *Journal of Machine Learning Research*, 3:1107–1135, 2003.

[3] K. Barnard and Q. Fan. Reducing correspondence ambiguity in loosely labeled training data. In *IEEE CVPR*, 2007.

[4] K. Barnard, Q. Fan, R. Swaminathan, A. Hoogs, R. Collins, P. Rondot, and J. Kaufhold. Evaluation of localized semantics: data, methodology, and experiments. *IJCV*, 77:199–217, 2008.

[5] K. Barnard and D. Forsyth. Learning the semantics of words and pictures. In *International Conference on Computer Vision*, pages II:408–415, 2001.

[6] P. Carbonetto, N. d. Freitas, and K. Barnard. A statistical model for general contextual object recognition. In *ECCV*, volume I, pages 350–362, 2004.

[7] T. Deselaers, B. Alexe, , and V. Ferrari. Localizing objects while learning their appearance. In *ECCV*, volume 6314 of *LNCS*, pages 452–466. Springer, 2010.

[8] H. J. Escalante, C. A. Hernandez, J. A. Gonzalez, A. Lopez-Lopez, M. Montes, E. F. Morales, L. E. Sucar, L. Villasenor, and M. Grubinger. The segmented and annotated iapr tc-12 benchmark. *Computer Vision and Image Understanding*, 114(4, Special issue on Image and Video Retrieval Evaluation):419–428, 2010.

[9] C. Fellbaum, P. G. A. Miller, R. Tengi, and P. Wakefield. Wordnet - a lexical database for english.

[10] P. Felzenszwalb, R. Girshick, D. McAllester, and D. Ramanan. Object detection with discriminatively trained part-based models. *IEEE PAMI*, 2009.

[11] A. Gupta and L. S. Davis. Beyond nouns: Exploiting prepositions and comparative adjectives for learning visual classifiers. In *In ECCV*, 2008.

[12] Y. Jin, L. Khan, L. Wang, and M. Awad. Image annotations by combining multiple evidence & wordnet. In *ACM MM '05*, New York, NY, USA, 2005.

[13] V. Lavrenko, R. Manmatha, and J. Jeon. A model for learning the semantics of pictures. In *NIPS*, 2003.

[14] J. Shi and J. Malik. Normalized cuts and image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(9):888–905, 2000.

[15] D. M. Tax, M. V. Breukelen, R. P. Duin, and J. Kittler. Combining multiple classifiers by averaging or by multiplying?, 2000.

[16] J. Verbeek, M. Guillaumin, T. Mensink, and C. Schmid. Image annotation with tagprop on the mirflickr set. In *MIR '10*, pages 537–546, New York, NY, USA, 2010. ACM.

[17] H. Zhang, A. C. Berg, M. Maire, and J. Malik. Svm-knn: Discriminative nearest neighbor classification for visual category recognition. In *In CVPR*, pages 2126–2136, 2006.

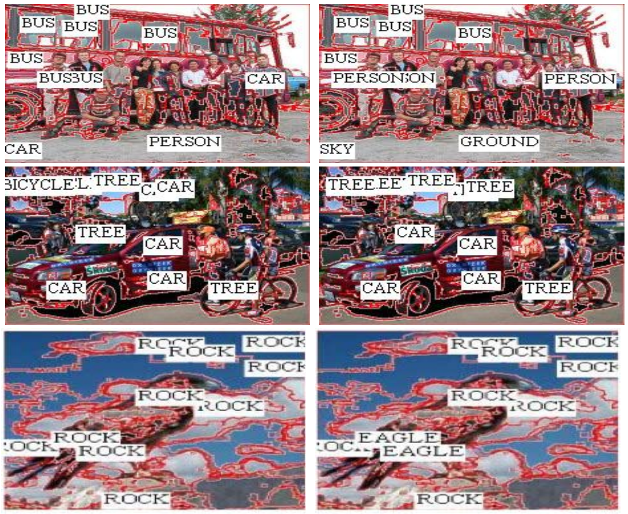

Figure 4: Adding detectors (right) improves the baseline algorithm (left). For better visibility, we only show the labels for the 10 largest regions