Top-Down and Bottom-up Cues for Scene Text Recognition

Vincent BODIN & Thomas MOREAU

December 16, 2013

Introduction

- Robust methods have allowed recognizing letters efficiently recently (OCR).
- Scene text recognition has been a center of interest in computer vision for those few last years.
- We decided to implement a paper by A. Mishra, K. Alahari, C. V. Jawahar, *Top-down and Bottom-up Cues for Scene Text Recognition* [1]

Sommaire

Learning characters

We need to build classifiers to recognize characters in a natural picture:

1. Use a database to identify characters: ICDAR 2003 [2], Chars74K [3];
2. extract features: Histogram Of Gradient (HOG) [4][1];
3. build K SVMs (K is the number of classes, 62) with RBF Kernel:

$$\exp(-\gamma|x - x'|^2), \gamma > 0 \tag{1}$$

method one-versus-all, we used a Python librairy scikit-learn [5]: two parameters γ and regularization C optimized by cross-validation -> test error without optimization 25%, with optimization error.

[1] N. Dalal, B. Triggs, Histogram of oriented gradients for human detectection

Learning words

We have to build a prior lexicon that contains how characters interact to create words.

1. Load a database of words [6][2];
2. count for each pair of characters (c_i, c_j) their frequency of occurrence $p(c_i, c_j)$ in the database (bi-gram model);
3. will be used as an energy term to be minimized afterwards.

[2] `http://algoval.essex.ac.uk/icdar/Datasets.html`

Sommaire

Sliding window and pruning

- Characters retrieved by sliding windows of different scales scanning the image;
- for each window l_i, compute the features ϕ_i of HOG with 12 orientations;
- run the 62 SVMs on ϕ_i and compute the goodness score:

$$GS(l_i) = \max_c p(c|\phi_i) \exp\left(-\frac{(a_i - \mu_{a_j})^2}{2\sigma_{a_j}^2}\right) \tag{2}$$

where a is the aspect ratio and j is the class that reaches the maximum *i.e.* the index of the character that has the highest probability of being in this window.

Sliding window and pruning

- if $GS(l_i) > 0.1$ then we keep the window *and* all the probabilities of each character.
- apply Non-Maximum Suppression algorithm (NMS) to merge windows that overlap:
  1. find the window with maximum confidence, say l_i.
  2. for all others l_j compute:

$$\text{criterion} = \frac{l_i \cap l_j}{l_i \cup l_j} \tag{3}$$

  3. if criterion $>$ `threshold` then merge the windows.
  4. the ending merged window has the highest confidence and is located at the barycenter of all the windows merged.

Sommaire

Graph construction

This part of the paper is implemented for the course of PGM, we will be brief!

Suppose we are given a set of n windows with possible characters inside.

1. for each sliding window assign a node that takes values in $\mathcal{K}_\epsilon^n$ (add ϵ void label);
2. build edges if the windows are 'close enough';
3. assign unary energy to each node, and pairwise energy for edges composed of a term of overlapping and lexicon prior;
4. minimize the discrete energy on the graph (NP-hard) with TRW-S algorithm [7].

References

C. V. Jawahar A. Mishra, K. Alahari.
Top-down and bottom-up cues for scene text recognition.
IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2012.

Robust ocr.
http://algoval.essex.ac.uk/icdar/Datasets.html.
Accessed: 2013-12-16.

The chars74k dataset: Character recognition in natural images.
http://www.ee.surrey.ac.uk/CVSSP/demos/chars74k/.
Accessed: 2013-12-16.

B. Triggs N. Dalal.
Histogram of oriented gradients for human detectection.
CVPR, 2005.

Scikit-learn.
http://scikit-learn.org/stable/modules/svm.html#parameters-of-the-rbf-kernel.
Accessed: 2013-12-16.

Robust word recognition.
http://algoval.essex.ac.uk/icdar/Datasets.html.
Accessed: 2013-12-16.

V. Kolmogorov.
Convergent tree-reweighted message passing for energy minimization.
IEEE Transactions on Pattern Analysis and Machine Learning (PAMI), 2006.