Paper Gestalt

Carven von Bearnensquash
Department of Computer Science
University of Phoenix
bearensquash@live.com

Abstract

Peer reviews of conference paper submissions is an integral part of the research cycle, though it has unknown origins. For the computer vision community, this process has become significantly more difficult in recent years due to the volume of submissions. For example, the number of submissions to the CVPR conference has tripled in the last ten years. For this reason, the community has been forced to reach out to a less than ideal pool of reviewers, which unfortunately includes uninformed junior graduate students, disgruntled senior graduate students, and tenured faculty. In this work we take the simple intuition that the quality of a paper can be estimated by merely glancing through the general layout, and use this intuition to build a system that employs basic computer vision techniques to predict if the paper should be accepted or rejected. This system can then be used as a first cascade layer during the review process. Our results show that while rejecting 15% of "good papers", we can cut down the number of "bad papers" by more than 50%, saving valuable time of reviewers. Finally, we fed this very paper into our system and are happy to report that it received a posterior probability of 88.4% of being "good".

1. Introduction

Peer reviews of conference paper submissions is an integral part of the research cycle, though it has unknown origins. For the computer vision community, this process has become significantly more difficult in recent years due to the volume of submissions. For example, the number of submission to the CVPR conference has tripled in the last ten years[1] (see Fig. 1). For this reason, the community has been forced to reach out to a less than ideal pool of reviewers, which unfortunately includes uninformed junior graduate students, disgruntled senior graduate students,

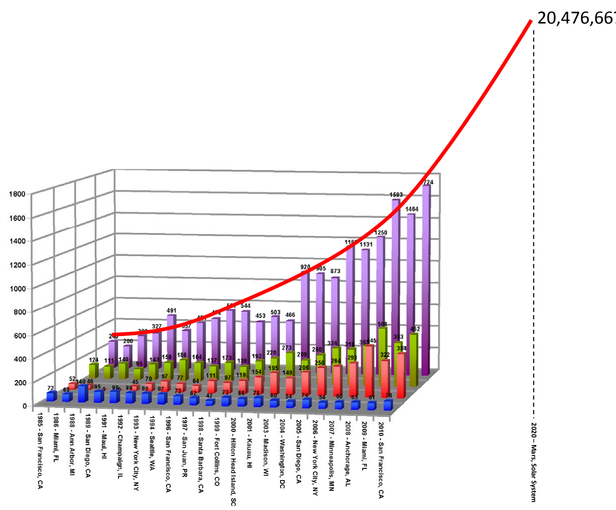

Figure 1. **Paper submission trends.** The number of submitted papers to CVPR, and other top tier computer vision conferences, is growing at an alarming rate. In this paper we propose an automated method of rejected sub-par papers, thereby reducing the burden on reviewers.

and tenured faculty. Although many excellent research papers have been published in the area of computer vision [3, 14, 15, 11, 19, 8, 7, 16, 1, 18, 17, 5, 2, 20, 22, 13], many good papers are rejected and many bad papers are accepted due to the imperfect review process.

In this work we take the simple intuition that the quality of a paper can be estimated by merely glancing through the general layout, and use this intuition to build a system that emplys basic computer vision techniques to predict if the paper should be accepted or rejected. We call the set of visual features that have discriminative power in this task the "paper gestalt". To build our system, we use powerful statistical learning techniques [4].

The rest of the paper is laid out as follows. In Section 2 we discuss some of the related work (though our work is so unique that there is little to discuss). In Section 3 we review our particular solution to this difficult problem. In Section 4 we present our thorough experimental results. Finally, in

[1] http://www.adaptivebox.net/CILib/CICON_stat.html.

Section 5 we conclude and point out promising directions for future research.

2. Previous Work

Basic, non-technical solutions are currently used to weed out poor quality submissions. An example is a forced registration and abstract submission deadline a week prior to the paper deadline, which eliminates spur-of-the-moment and half-assed submissions. Monetary incentives (*i.e.* a fee for submitting) have been discussed, but, to our knowledge, have never been implemented. Applying techniques from text processing, such as [21], could be considered as well. However, these techniques would analyze the text of the paper itself, while ignoring rich visual information. Furthermore, they could become biased to certain terms like "boosting", "svm", "context", or "crf" due to biased training data.

At the time of submission, we are unaware of any previous work that attempts to automate the reviewing process via computer vision technology. Unlike text based methods, our approach is able to capture rich visual information; furthermore it preserves privacy by ignoring the actual text of the article.

3. Approach

To solve this problem, we first formulate it as a binary classification task. We assume that we are given a training data set of example-label pairs, $\{(x_1, y_1), (x_2, y_2), ...(x_n, y_n)\}$, where $x_i \in \mathcal{X}$ is a vector of feature values computed for example paper i, and $y \in \{0, 1\}$ is a binary label of example i (in our case positive examples are papers that are "good" or should be accepted, and negative examples are papers that are "bad" and should be rejected). The goal is to then learn a function $f : \mathcal{X} \rightarrow \{0, 1\}$. An overview of the training procedure is shown in Figure 2. Our system is built similar to that of [19]; we give details below.

3.1. Learning Algorithm

We chose AdaBoost [10] as the learning algorithm for our system. The form of the classifier is as follows:

$$h(x) = \sum_{t=1}^{T} \alpha_t h_t(x) \tag{1}$$

where h_t is a weak classifier. The equation above returns a confidence score. As is commonly done, we use a decision stump as the weak classifier:

$$h_t(x) = \mathbf{1}[f_t(x) > \theta] \tag{2}$$

- Given example images $(x_1, y_1), \ldots, (x_n, y_n)$ where $y_i = 0, 1$ for negative and positive examples respectively.
- Initialize weights $w_{1,i} = \frac{1}{2m}, \frac{1}{2l}$ for $y_i = 0, 1$ respectively, where m and l are the number of negatives and positives respectively.
- For $t = 1, \ldots, T$:
  1. Normalize the weights,
  $$w_{t,i} \leftarrow \frac{w_{t,i}}{\sum_{j=1}^{n} w_{t,j}}$$
  so that w_t is a probability distribution.
  2. For each feature, j, train a classifier h_j which is restricted to using a single feature. The error is evaluated with respect to w_t, $\epsilon_j = \sum_i w_i \, |h_j(x_i) - y_i|$.
  3. Choose the classifier, h_t, with the lowest error ϵ_t.
  4. Update the weights:
  $$w_{t+1,i} = w_{t,i} \beta_t^{1-e_i}$$
  where $e_i = 0$ if example x_i is classified correctly, $e_i = 1$ otherwise, and $\beta_t = \frac{\epsilon_t}{1-\epsilon_t}$.
- The final strong classifier is:
$$h(x) = \begin{cases} 1 & \sum_{t=1}^{T} \alpha_t h_t(x) \geq \frac{1}{2} \sum_{t=1}^{T} \alpha_t \\ 0 & \text{otherwise} \end{cases}$$
where $\alpha_t = \log \frac{1}{\beta_t}$

Figure 3. **Psuedo-code for AdaBoost** (figure reproduced from [19]).

where θ is a threshold and f_t is some scalar feature of the image. Below we review some of the features that we implemented in our system. The training procedure for this algorithm is summarized in Figure 3.

AdaBoost has many appealing theoretical properties. For example, it is well known that the empirical error is bounded [9]:

$$\epsilon(h) \leq \prod_{t=1}^{T} 2\sqrt{\epsilon_t(1-\epsilon_t)} \tag{3}$$

Though this does not affect our system directly, we have found that equations improve paper gestalt and thus increase the chances of this paper being accepted (*c.f.* Figure 6). While this is a good start, we believe that the amount of math in this paper is still not adequate (*c.f.* Figure 6). Therefore, purely for aesthetic purpose, we reproduce Maxwell's equations below [12]:

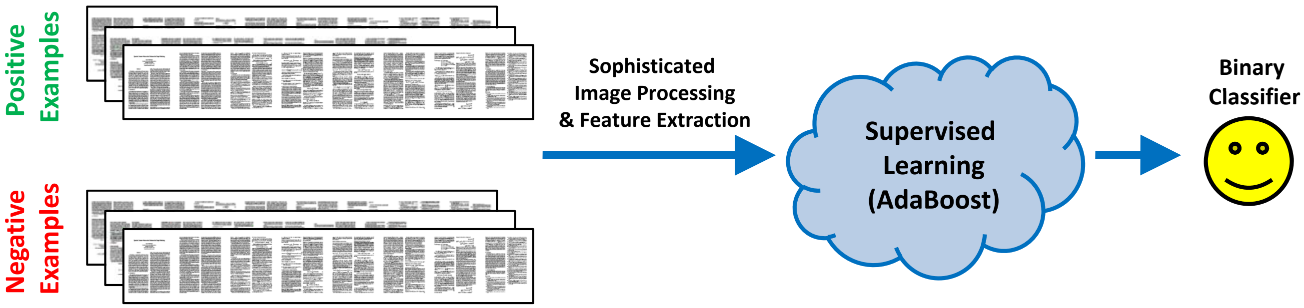

Figure 2. **Overview of the training procedure.**

$$\oint_{\substack{\text{closed}\\\text{surface}}} \vec{E} \cdot d\vec{A} = \frac{Q_{enc}}{\epsilon_0} \tag{4}$$

$$\oint_{\substack{\text{closed}\\\text{surface}}} \vec{B} \cdot d\vec{A} = 0 \tag{5}$$

$$\oint \vec{E} \cdot d\vec{s} = -\frac{d\phi_B}{dt} \tag{6}$$

$$\oint \vec{B} \cdot d\vec{s} = \mu_0 \epsilon_0 \frac{d\phi_E}{dt} + \mu_0 i_{enc} \tag{7}$$

3.2. Features

Given an image of a paper, we need to compute a number of visual features that can be plugged into the classification system. We chose a number of standard computer vision features that capture gradient, texture, color and spatial information. In particular, we compute features based on LUV histograms, Histograms of Oriented Gradients [6] and gradient magnitude.

4. Experiments and Results

4.1. Data Acquisition

It is well known that choosing a good dataset is vital for publication [13]. To train our classifier and evaluate its performance we first needed to collect a dataset of good and bad papers (*e.g.* positive and negative examples). We gathered accepted papers from the following top tier conference proceedings: CVPR 2008, ICCV 2009, and CVPR 2009. Since there was no way for us to collect papers that were rejected from these conferences, we instead collected the workshop papers from these same conferences as an approximation. Note that the format is the same for all of the papers we collected. Our dataset consisted of 1196 positive examples and 665 negative examples.

We converted all papers from pdf format into images by concatenating the pages, and then resized the images to 1132 by 200 pixels. Papers that were less than 8 pages in

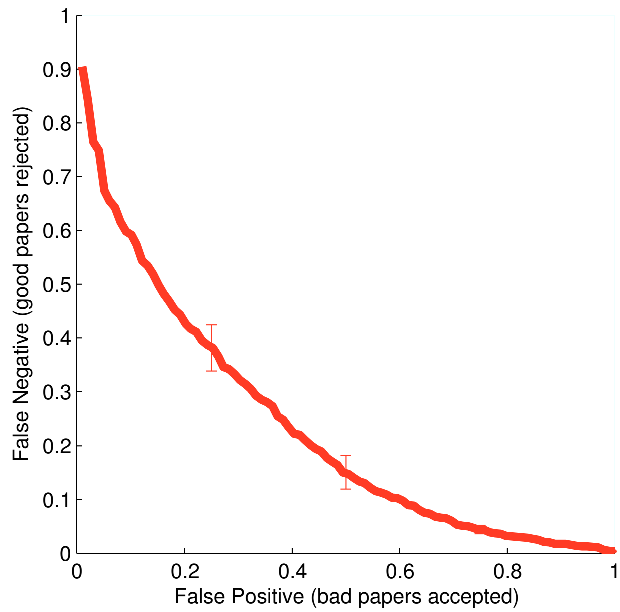

Figure 4. **Obligatory ROC curve.** The plot is computed by sweeping of a confidence threshold. Note that if we allow our classifier to reject 15% of good papers, we can throw out half of the bad papers, dramatically reducing the amount of time reviewers have to spend reading bad papers.

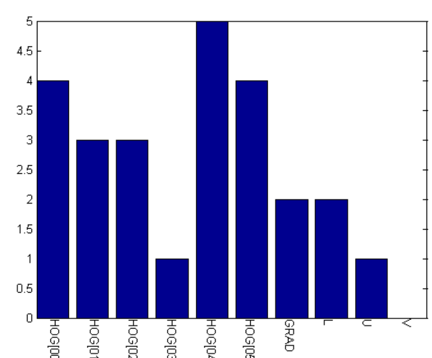

Figure 5. **Histogram of feature usage.**

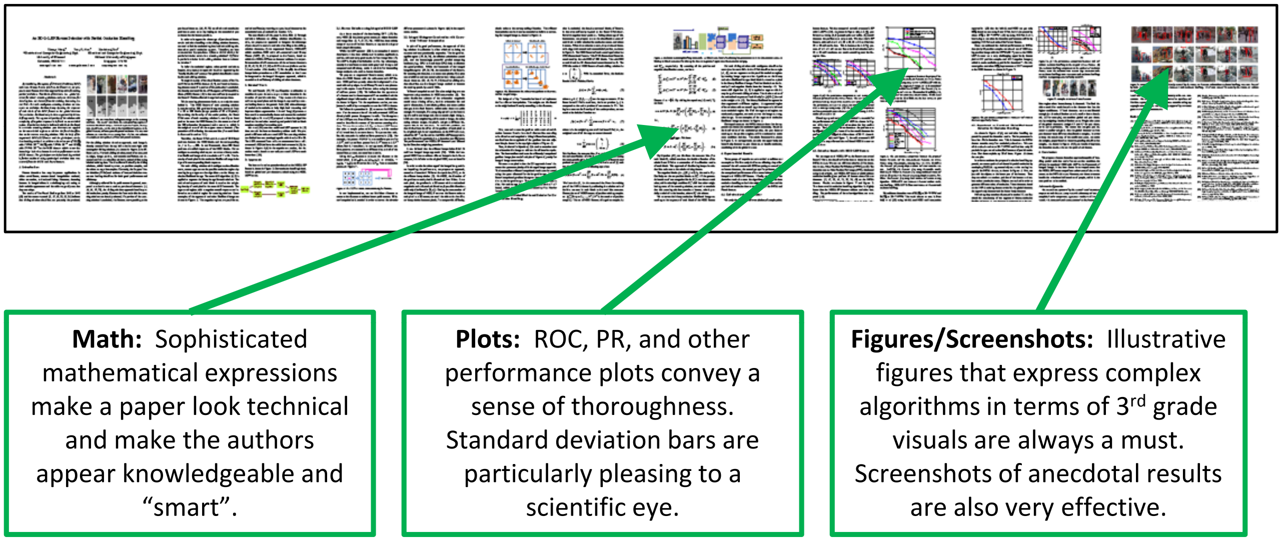

Figure 6. **Characteristics of a "Good" paper.**

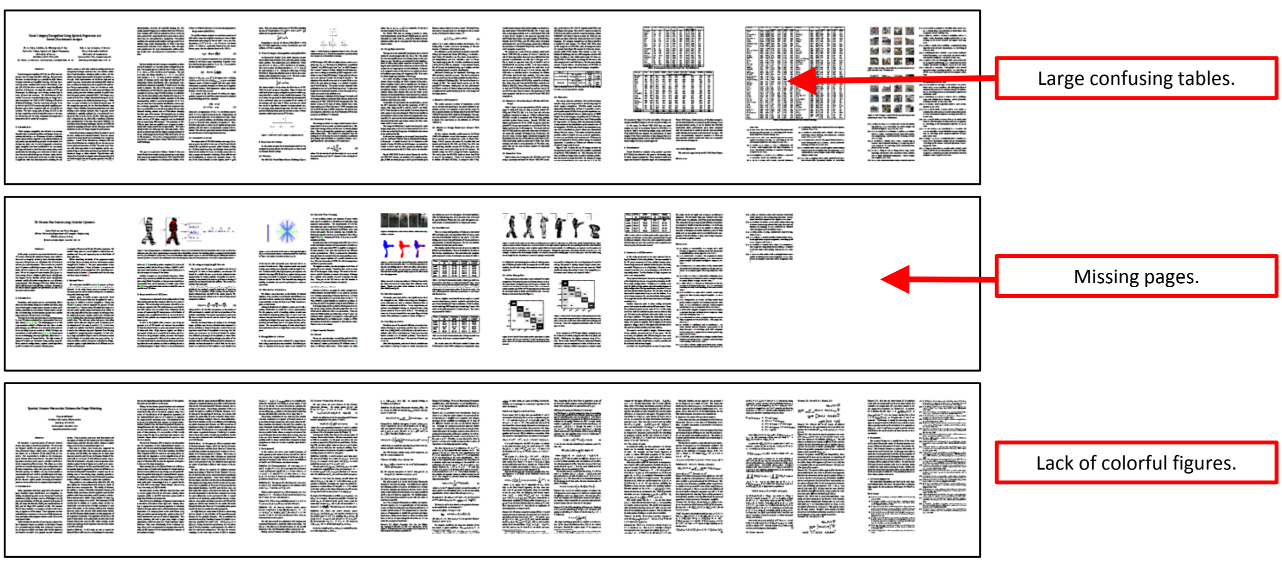

Figure 7. **Characteristics of a "Bad" paper.**

length were padded with blank pages, so that the images were all of the same size.

4.2. Performance Evaluation

We randomly split the data into 25% testing and 75% training. All the reported results are averaged over 5 trials with different random splits. Figure 4 shows the ROC curve of our classifier. We computed this curve by sweeping over confidence values of the classifier. The main result of this paper is as follows. *If we assume that rejecting 15% of good papers is acceptable since human reviewers tend to make mistakes anyway, our system can throw out more than 50% of bad papers, cutting the workload of reviewers in half.*

4.3. Analysis

Let us now take a closer look at what types of visual characteristics distinguish a good paper from a bad paper. First, in Figure 5 we plot a distribution of features that were chosen by the boosting algorithm. Though admittedly, we're not sure what this figure reveals, we believe that bar plots are particularly aesthetically pleasing.

Next, in Figure 6 we highlight certain visual characteristics that we have noticed among the set of good papers. Similarly, in Figure 7 we highlight characteristics of bad

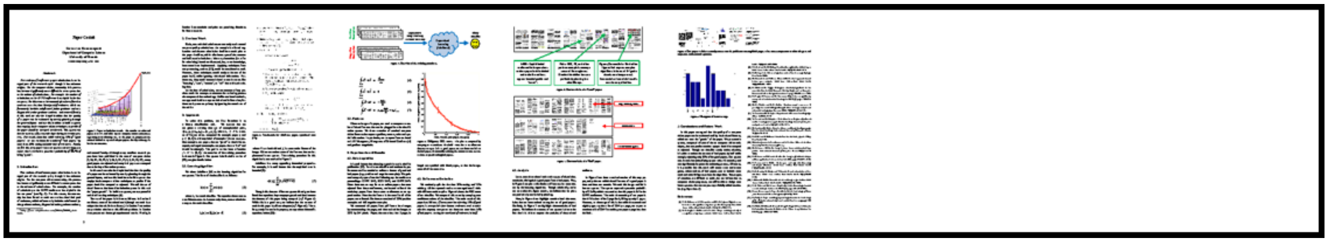

Figure 8. **Our paper**. While it certainly suffers from the problem of missing/blank pages, it has a nice composition of colorful figures and impressive mathematical equations.

papers. We believe the success of our system is due to the fact that it is able to capture the statistics of these visual attributes.

In Figure 8 we show a resized version of this very paper, and point out which visual features of a good paper it does and does not contain. We took this image and fed it into our system. The system reported a posterior probability of 88.4%, which reassured us that this paper is fit for the CVPR conference. The main shortcoming of our paper is that it falls short of the 8 page limit, filling up only 5 pages. However, an advantage of this is that while the seventh and eighth pages require a fee of \$100 per page, we expect to receive credit of \$100 for making our paper a page less than the limit.

5. Conclusions and Future Work

In this paper we argued that the quality of a computer vision paper can be estimated well by basic visual features, which we term the "gestalt" of the paper. We presented a system, composed of state of the art computer vision techniques, that can predict weather a paper should be accepted or rejected. Though our classifier surely makes mistakes, it is able to reduce the amount of bad papers by half while wrongly rejecting only 15% of the good papers. Our system runs in real-time (classifying a paper takes 0.5 seconds), and can significantly improve the reviewing process. Of course, it is possible that this work will create a cat and mouse game, where authors of bad papers start to include more math and colorful figures to beat the algorithm. These types of situations are difficult to avoid, but we believe that as computer vision progresses, we will be able to design even better systems that can see past superficially added aesthetics (*e.g.* Equations 4).

References

[1] P. N. Belhumeur, J. P. Hespanha, and D. J. Kriegman. Eigenfaces vs. fisherfaces: Recognition using class specific linear projection. *IEEE Trans. Pattern Anal. Mach. Intell.*, 19(7):711–720, 1997.

[2] S. Belongie, J. Malik, and J. Puzicha. Shape matching and object recognition using shape contexts. *IEEE Trans. Pattern Anal. Mach. Intell.*, 24(4):509–522, 2002.

[3] P. J. Besl and N. D. McKay. A method for registration of 3-d shapes. *IEEE Trans. Pattern Anal. Mach. Intell.*, 14(2):239–256, 1992.

[4] C. Bishop et al. *Pattern recognition and machine learning*. Springer New York:, 2006.

[5] D. Comaniciu and P. Meer. Mean shift: A robust approach toward feature space analysis. *IEEE Trans. Pattern Anal. Mach. Intell.*, 24(5):603–619, 2002.

[6] N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, volume 1, page 886. Citeseer, 2005.

[7] R. Fergus, P. Perona, and A. Zisserman. Object class recognition by unsupervised scale-invariant learning. In *CVPR (2)*, pages 264–271, 2003.

[8] M. A. Fischler and R. C. Bolles. Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography. *Commun. ACM*, 24(6):381–395, 1981.

[9] Y. Freund and R. Schapire. Experiments with a new boosting algorithm. In *ICML*, 1996.

[10] Y. Freund and R. Schapire. A decision-theoretic generalization of on-line learning and an application to boosting. *Journal of computer and system sciences*, 55(1):119–139, 1997.

[11] M. Isard and A. Blake. Condensation - conditional density propagation for visual tracking. *International Journal of Computer Vision*, 29(1):5–28, 1998.

[12] C. Kittel and P. McEuen. *Introduction to solid state physics*. Wiley New York, 1996.

[13] D. LaLoudouana, L. Tecallonou, and J. Puzicha. Data set selection. *Journal of Machine Learning Gossip*.

[14] D. G. Lowe. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 60(2):91–110, 2004.

[15] B. D. Lucas and T. Kanade. An iterative image registration technique with an application to stereo vision. In *IJCAI*, pages 674–679, 1981.

[16] P. Perona and J. Malik. Scale-space and edge detection using anisotropic diffusion. *IEEE Trans. Pattern Anal. Mach. Intell.*, 12(7):629–639, 1990.

[17] H. A. Rowley, S. Baluja, and T. Kanade. Neural network-based face detection. *IEEE Trans. Pattern Anal. Mach. Intell.*, 20(1):23–38, 1998.

[18] J. Shi and J. Malik. Normalized cuts and image segmentation. *IEEE Trans. Pattern Anal. Mach. Intell.*, 22(8):888–905, 2000.

[19] P. A. Viola and M. J. Jones. Rapid object detection using a boosted cascade of simple features. In *CVPR (1)*, pages 511–518, 2001.

[20] C. von Bearensquash. Paper Gestalt. In *Secret Proceedings of Computer Vision and Pattern Recognition (CVPR)*, 2010.

[21] H. Wallach. Topic modeling: beyond bag-of-words. In *Proceedings of the 23rd international conference on Machine learning*, page 984. ACM, 2006.

[22] R. White, O. Arikan, H. Arora, A. Berg, T. Berg, J. Chan, J. Edwards, A. Farhadi, L. Ikemoto, N. Ikizler, et al. Where is my Advisor? In *CVPR*, pages 148–156, 2006.