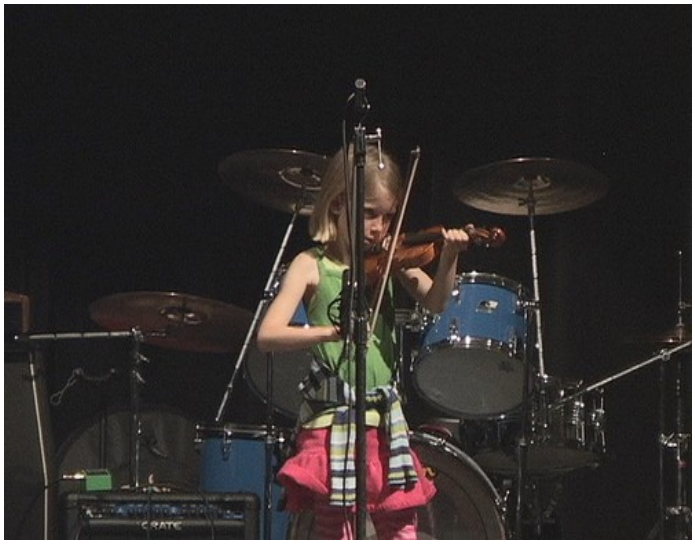
Identifying Iconic Images of Objects from Tagged Internet Picture Sets

Prasad Tare

1. Introduction

Although there may be thousands of images that contain a certain object somewhere in them, the majority of these images will show the object obscured by other things in the picture, placed in relation with them, or in any other way that makes it unclear what the central object is. Some of the images, however, will show the object clearly separated from the background and in a form that is typical of it. We call the most typical examples of an object iconic images. For example, of the thousands of pictures of violins, the majority will show the violin either being handled by a person, with many other objects obscuring it, or will be zoomed in on a component (show just the bow, or just the strings). The two pictures below illustrate this difference. The picture on the left is an iconic image of the violin, but the picture on the right, while it is correctly classified as a violin, also holds many other objects (and a violinist). For consistency, most of the images shown as examples will be drawn from the violin set.



The existence of iconic images in human perception is supported by psychological evidence [1]. Humans recognize such images faster as belonging to a certain category. Such pictures could also be used to teach a child, or someone learning the language, about that object. This project applied machine learning and image processing to identify the iconic images from picture sets, given that each picture contains the object in some form. Finding such images can lead to much larger image category databases at lower human cost. While some researchers in this field suggest it, I do not believe that this will have any applications to ranking search engine results of images, because it doesn't deal with frequencies of viewing, nor does it incorporate any textual clues. Also, it is not essential to identify each and every iconic image from a given set.

2. Data

The picture sets for this project are obtained from Image Net (www.image-net.org). I worked with six data sets: violins (fiddles), horses (saddle horses), hand calculators , doors (swinging or sliding

barrier), sinks, and table lamps. To keep it computationally manageable, 1 000 images were randomly selected from each set. As a direct application, iconic images could be used to order the results in Imagenet.

3. Current Work

Work has been done in sifting through images from internet photo archives such as Flickr to identify iconic images [2]. Their work was largely the inspiration for my project, and formed the structure and informed my approach. After preprocessing the images, I tried different approaches (different machine learning algorithms) to improve the results. The basic features and object detection steps were taken from their paper.

4. Approach

The overall approach can be divided into two stages, each reflecting part of the definition of an iconic image. First, the object should be clearly visible and separated from background, so first we need to recognize the images which contain such a depiction. Object detection is a nontrivial problem in computer vision, and is certainly not my area of expertise, but without doing this step, there are far too many images to apply other techniques efficiently to, and other features in the data start to be fitted more. In my preliminary testing, with the violin data set, the algorithm starts to cluster around the persons or other such objects in the images, because often the violins are much smaller. Object detection can help filter those out. We should immediately discard images that depict multiple possible objects, because such images cannot be iconic anyway. These are referred to as “junk” or “distractors”.

The second step assumes that we now have a much smaller set of images that, at the minimum show, an object clearly. From these, we wish to identify the most typical ones. The biggest challenge is to find the relevant features to look at. After we have found those features, the iconic images should all be similar, so we need to find modes in the data, by running an unsupervised clustering algorithm. Another possible approach is to use Gaussian Discriminant Analysis (GDA) with the labels being the six different categories (ex: violins, horses,etc). After training the probability distribution, we can extract the images most likely to be violins, and perhaps these will be the most typical (iconic) violins.

5. Object Detection

To detect images with a clear object, I went through multiple data sets (not all of them are the same category, this is a more general problem), and hand labeled images that contain objects clearly,

and also marked a rectangle around the object. I also produced some negative training data by incorrectly labeling a few. For example, the above image depicts an object clearly (though it is not an iconic image). Literature suggests the following feature to use: hue, saturation, value, focus, and texture. These features are computed on the object rectangle (where we have tagged the object), and also on the remaining image (the background). For the hue, saturation, and value, I used the distribution of pixel values (binned into a histogram) as the feature vector. It is very hard to determine where the object lies, but not too inefficient (surprisingly) to iterate over multiple possible locations of rectangles for the object, and compute the features for background and foreground. A Naïve Bayes algorithm

learns the distribution of the features on the background and foreground, and as we iterate over the foreground rectangle positions, selects the one that has the maximum probability of containing an object. After running this over all the images, the images that don't contain a clear object will be removed. The training data was relatively small (~75 hand labeled images), but this was enough to go on to the next part.

Some of the issues are that image regions that are very different are marked as objects, which means that the sky, whenever it appears, always gets marked as the object, but this is clearly undesirable. This led to slightly worse performance on the horse data set.

6. Selecting Iconic Images

We note that the local appearance of a violin (or horse, or any category) is going to remain fixed: in different images, whether they be iconic or not, the position, camera angle, illumination, and color will change. Thus, as our features, we don't want to capture hue or intensity, for example, but rather we want to capture the underlying geometric shape. The traditional methods, such as cross correlation, will not work because we specifically know that changes in illumination do not indicate differences. Shape descriptors are needed. This is far from a solved problem, and one insight I drew from the literature is that, in order to make out algorithm robust to distortion, one way is to blur the pixels: this can be easily accomplished by convolving with a Gaussian (which in turn can be most easily accomplished by taking the 2D fft and multiplying, and then doing ifft). A somewhat similar problem, object recognition using template matching, uses a more sophisticated method called geometric blur descriptors to achieve this, and these are the features that I chose to use. From a high level view, they involve convolving the image, which we wish to compute features for, with a spatially varying Gaussian. The parameters of the filter, as well as the number of sample points to take, all were determined to allow the feature vector to capture the geometry we wish to compare between images.

An open source library VGG provided routines for computing these (http://www.robots.ox.ac.uk/~vgg/software/MKL), although in the time in took to get the library to work, I could have written my own. More details about geometric blur descriptors can be found in the references, which are the papers I learned about them from [3]. These features were used for all the machine learning algorithms tried.

This is an unsupervised learning problem. Based on the intuition that all the iconic images should look similar to each other (after all, they must be typical of that category), the first algorithm implemented was k means to detect modes in the data. The centroids of the clusters that contain many images close to them will be taken to be iconic images. A different way of identifying the clusters would be to adopt a probabilistic approach and use mixture of Gaussians. For this, I defined the latent variables as being different levels of iconicity, and it makes sense that the distribution of features depends on how iconic the image is. If we have an iconic image of violin, the features will, with high probability, be similar to each other, but quite different from the features of just a random image that contains a small violin in the corner.

The third approach I tried was to solve a related supervised learning problem, with the class labels being the different object categories. A Gaussian Discriminant Analysis (GDA) model was trained to this data, and the intuition was that this allows us to build a model of what a violin looks like, what a horse looks like, etc. When the model is rerun, treating the images as the test data, the images with the highest probability of belonging to that category are taken to be the iconic images. Under the assumption that the tags (labels A and B) are correct, I used the Bayes' Net Toolbox to train a GDA model between different class labels: violins, calculators, horses, doors, sinks, and lamps. This

approach is promising because it can be applied to new images that were not part of the initial set, after we have learned the parameters.

7. Results

The object detection part of the project performs relatively well in that the images that it identifies as containing a distinct object do indeed contain a distinct object, and the foreground (object)



rectangle is correctly drawn. Besides the top few images, the algorithm performs poorly in detecting objects, so it isn't a solution to the general object detection problem, but it works for this case because the iconic images are typically very clear.

A major pitfall of this project is that there is no way to quantify how close we have come to the right answer, and I relied on looking at the images suggested by the algorithm, and then subjectively decided whether they are correct or not. It is also awkward to present 120 images in a paper. With that disclaimer, k means clustering performs surprisingly well and identifies images that are the iconic images. The image on the first page was one recognized by k means, for example. The best performance is obtained on categories like calculators or violins, which have a very distinctive shape. In the data sets, one cluster emerges with 30 or 40 images very close to the centroid, while the other clusters are more diffuse. Most of the images that are very close to that centroid are images that are recognized as the iconic images. There are some errors produced by consistent errors (biases) in the data. For example, there are many pictures of people holding violins, and secondary clusters tend to gather around those. Also, the performance is worse in the lamp data set, because there are many different shapes of lamps. In that case, three clear clusters emerged, and the images close to the centroid of each were all iconic images of lamps (each cluster, however, was quite different from the others, reflecting different “types” of the category lamp). Even in that case, the majority of the images wind up in diffuse clusters that can be discarded as not being iconic. This suggests that this approach will not work well for a very general category, the objects in which have a varied geometric appearance. For a specific object, however, k means performs very well.

The performance of mixture of Gaussians (MG) is fairly similar to that of k-means. In the horse data set, for example, 11 of the top 20 iconic images generated by MG matched those from k means, and an additional 4 images also being acceptable as iconic, but the remaining 5 were “distractors.” In some data sets, MG does well, and it's not clear at this point why. It doesn't seem that the probabilistic approach is the correct model for this problem, because the non probabilistic clustering algorithm k means has superior performance. In the future work section, I discuss a way to leverage both methods together.

The GDA approach to the problem, however, performs very poorly. It seems that modeling a violin as “not a horse, calculator, or lamp” is not an effective way to recognize the most typical images of violins. Without a focus group to evaluate the results by ranking the images and identify what types of errors are being made, it is also not clear how or if GDA can be tweaked in some way to improve its

performance. My intuition suggests that picking two categories with very similar geometric appearance (ex: horses and donkeys) may yield slightly better results, but without diagnostics, it's not clear that this should be the next step. Given that k means works fairly well, the problem can be handled in the framework of unsupervised learning, and an algorithm like GDA is not needed. Also, the complexity of GDA increases with the number of categories because it compares between them: k means and MG both operate only with images tagged as a certain category. Hence, GDA runs fairly slowly because it has to deal with a large number of images. This approach seemed promising because the parameters can be learned from a subset of the images, and then applied to the rest, but the performance is too poor to explore this further.

8. Future Work

As mentioned above, the most important thing in the future would be a way to verify the results than the opinion of one researcher: a group of people who ranked the images, or indicated with the identification of a certain image as being iconic is correct, would be very helpful in quantifying the performance, a resource which some researchers in the field had [2].

A possible avenue for future research would be combining MG and k means. MG can be used for anomaly detection, and this may allow us to reject consistent distractors (people in the violin set), which will enhance the performance of k means.

9. Conclusion

A method for automatically detecting iconic images was implemented, and the results were surprisingly good. Most of the images identified as "iconic" were indeed typical of that category. Object detection is a crucial step for discarding unfocused images, without a clear object or with many objects in them. Geometric blur descriptors are a good choice for the features in this problem. K means performs the best in identifying modes in the data, although the performance of MG is similar in some cases, but much worse in others. GDA does not seem to be the right approach to this problem. Despite not having a means of quantifying the results, this project implemented the state of the art, and moved beyond it by trying other promising approaches, and rejecting one entirely (GDA). I would like to acknowledge the help of Prof. Fei Fei Li in this project.

References

1. Palmers, S. et al. *Canonical perspective and the perception of objects.* Attention and Performance IX 1981.
2. Berg, Tamara and Alexander Berg. *Finding Iconic Images*. Internet Vision Workshop (CVPR 2009).
3. Berg, Alexander and Jitendra Malik. *Geometric Blur for Template Matching.* Computer Vision and Pattern Recognition, 2001. CVPR 2001.
4. Chen, Bo et al. *Human Pose Estimation with Rotated Geometric Blur.* IEEE Workshop on Applications of Computer Vision (WACV), 2008.