

Image Classification with Annotator Rationales



Jeff Donahue

Supervised by Professor Kristen Grauman

Department of Computer Science

The University of Texas at Austin

A thesis submitted for the degree of
Bachelor of Science - Turing Scholars Honors

December 2010

Abstract

In image classification tasks, the traditional supervised learning approach is for annotators to simply label each image with its class name. We contend that this approach may waste potentially valuable information: the reasoning that went into the choice. This information can best be exploited in tasks where an element of subjectivity or perception is involved in the annotation. Hence, in a new approach, we enrich this simple categorical annotation by augmenting it with a “rationale”: a polygon drawn around the region(s) of the image that the annotator found most influential in his or her classification decision. This is distinct from foreground segmentation, as the entirety of the foreground may not have been influential to the annotator’s decision. To make use of this extra information, when creating a representation for the features of an image in the training set, we give special treatment to those features that fall inside of a rationale polygon. We have tested our approach on a scene classification task, with results showing that this extra bit of information is highly useful in deciding whether an image belongs to certain scene categories. We have further applied our approach to the more subjective task of deciding whether a person in an image is attractive, and have seen promising preliminary results in this domain as well.

Contents

List of Figures	vi
List of Tables	vii
1 Introduction	1
2 Related Work	4
2.1 Standard Mode of Training	4
2.2 Improving Annotation Effectiveness	5
2.3 Natural Language Processing	5
3 Approach	7
3.1 Image Representation	7
3.1.1 Scene Categories Data	8
3.1.2 Hot or Not Data	8
3.2 Training an SVM with Rationale Examples	9
3.3 Issues with the Rationale-Based Approach	12
3.3.1 Amenability to Different Tasks	12
3.3.2 Selection of Useful Rationales	14
4 Data	16
4.1 Scene Categories	16
4.2 Hot or Not	16

5	Annotations	20
5.1	Scene Categories	20
5.2	Hot or Not	23
6	Results	29
6.1	Scene Categories	29
6.1.1	Methodology	30
6.1.2	Baselines	30
6.1.3	Results of Scene Categories Experiment	31
6.1.4	Discussion	31
	6.1.4.1 Success of our Approach	31
	6.1.4.2 Intercategory Performance Disparities	34
6.2	Hot or Not	35
6.2.1	Methodology	35
6.2.2	Face Detection Baseline	36
6.2.3	Results of Hot or Not Experiment	37
6.2.4	Discussion	38
	6.2.4.1 Success of our Approach	38
	6.2.4.2 Differences in Classifying Males and Females	38
	6.2.4.3 Low Overall Performance	39
6.2.5	Classification Performance per Image	39
7	Conclusions	42
	References	43

List of Figures

1.1	Rationales for Various Tasks	2
3.1	Bag of Words Histogram Example	8
3.2	DoG vs. Dense Features	9
3.3	Original, Rationale, and Contrast Examples	10
3.4	SVM Modified for Contrast Examples	11
3.5	Algorithm Summary	13
3.6	Difficult Rationale Example	14
4.1	Fifteen Scene Categories - Sample Images	17
4.2	Hot or Not Interface	19
5.1	Normal Annotation Examples	22
5.2	Tight Annotation Examples	22
5.3	Artistic Annotation Examples	23
5.4	Pruning Examples	23
5.5	Hot or Not MTurk Interface	26
5.6	Hot or Not Annotations from MTurk	27
6.1	Scene Categories Precision-Recall Curves	33
6.2	Best and Worst Scene Categories	35

List of Tables

4.1	Hot or Not Data Statistics	18
5.1	Scene Categories MTurk Run Statistics	21
5.2	Human Confusion Matrix for Scene Categories	24
5.3	Hot or Not MTurk Run Statistics	28
6.1	Scene Categories Mean Average Precision	32
6.2	Hot or Not Results	37
6.3	Best Performance Improvement - Hot or Not	40
6.4	Worst Performance Loss - Hot or Not	41

1

Introduction

One of the most pervasive and fundamental challenges in computer vision today is that of visual classification. Visual classification has a wide variety of applications, such as automating the indexing of images for convenient retrieval, analyzing medical imagery, and data mining. There exists an immense body of research on methods to improve image classifier performance based on a set of training images and their class labels, but few papers have bothered to question the foundation on which such classifiers are based: the training data itself. In any supervised learning approach to image classification, an annotator will be shown an image and asked to put it in one or more categories based on its content, but this has generally been where the annotation stops. In this respect, current approaches are surprisingly inflexible in their means of collecting annotations for learning visual categories (e.g., of objects or actions). The standard approach of getting examples and category labels makes sense for learning a classifier, but we expect that human annotators can give us deeper cues in their annotation to better reveal the features that distinguish each category from the others.

We are interested in visual classification problems where a human can provide not only a categorization label, but also some insight into which aspects (in particular, which spatial regions) of the visual most helped the human determine that label. We believe that such information is not only useful, but is in fact *necessary* to learn the desired concept well for some problems - in particular, subjective and perceptual judgments (such as those about human emotions or expressions, a judgment of quality such as a rating given to an ice skating routine, etc.).

Thus, we propose a system in which the annotator not only indicates his or her category selection,



Is Brian Williams doing a serious story or soft news?



Is this scene from a comedy or a drama?



Is Mack Brown's team winning or losing?



Are these sitcom characters friends or nemeses?



Will Judge Judy rule for the plaintiff or the defendant?



How is this figure skater's form?

Figure 1.1: Rationales for Various Tasks - Examples of possible rationale annotations for various tasks. All of these tasks are subjective in nature (evaluating a general mood or attitude, or giving an opinion), and have relevant information confined to a certain portion of the image.

but also gives a rationale indicating the regions of the image that most influenced the selection by drawing polygons around these regions (figure 1.1). It is our contention that asking the annotator for this information can be used to focus the classifier on the features of the image that can truly be used to discriminate between two or more classes of images.

This approach is intuitively more valuable than current automated feature selection approaches (e.g., mutual information). Without injecting into a classifier knowledge of *why* a given classification was chosen, the classifier may draw false conclusions about the class based purely on an inadequate or biased set of training examples. For example, if we desire a classifier that distinguishes between images of a dog and images of a butterfly, and all of our training examples for the “dog” class are set indoors, whereas all of our “butterfly” examples contain a forest backdrop, the classifier may end up learning mainly the visual cues given by the backdrops, and misclassify images of these

animals in the opposite settings. This is, of course, an extreme example, but it illustrates the point that allowing human annotators to give rationales for their class selections could in effect give a classifier some of the information that a human annotator has naturally learned about the categories since birth, while an automated feature selection approach derives all of its evidence from the set of training examples alone. With that said, an automated feature selection method such as mutual information is potentially compatible with our approach; such a method would decide which types of visual information are the most valuable (i.e., which visual words to use), and our approach decides from which regions of the image to take visual information.

In order to make use of annotator rationales, our method uses a two-margin support vector machine (SVM), as in (1). As always, we want to maximize the margin between positive and negative examples, but in our case, we would also like to maximize the margin between positive examples and positive “contrast examples,” which include the image features that fell *outside* of any rationale polygon. Intuitively, if we remove the parts of the image that an annotator felt most influenced his or her class annotation, we should be less confident of its classification as a positive example, so we therefore make the SVM recognize this by adding a “contrast constraint,” creating a margin between positive examples and positive contrast examples.

This approach is a direct adaptation to computer vision of Zaidan, Eisner, and Piatko’s natural language processing work in (1), in which the authors show how they can improve the performance of an SVM that decides whether a movie review is positive or negative using their version of rationales: sequences of words that the annotator highlights.

We show how this learning algorithm can be adapted to the visual domain. We apply this algorithm to the Fifteen Scene Categories data set, demonstrating that it can be used to create stronger binary classifiers for many of the scene categories. Additionally, we create a new data set using images from a popular image rating website, showing how our approach can be used to enhance machine learning for a very subjective task: judging the appearance of a person as “Hot” or “Not.”

2

Related Work

There is a large body of work on standard methods of learning visual classes, methods of improving the effectiveness of human annotations, and analogous natural language processing work. We discuss some of it and contrast it with our work here.

2.1 Standard Mode of Training

Much work has been done using a standard mode of training for visual categorization systems, as exhibited by several benchmark visual data sets and collections (2, 3, 4, 5, 6, 7). Ordinarily, this work involves taking a set of images and associated class labels, forming a representation of these images based on local features, and using these image representations along with their class labels to train a classifier. The researchers then test this classifier by asking it to predict the class labels for some images it hasn't yet seen and comparing these predictions to ground truth. Annotations are generally fixed and uniform across examples, and the goal is nearly always to attain a subimage segmentation and a set of associated object names. These efforts will often begin with a keyword search to isolate candidate examples, followed by a thorough, human-controlled pruning to only those images or videos that truly show the category of interest.

In (8), Oliva and Torralba look at learning descriptors about scene images, such as *openness*, *ruggedness*, *naturalness*, *busyness*, and others. Lazebnik's work in (9) looks at using a pyramid match representation to get very high classification performance for a classifier over all fifteen scene categories. Like most work in image classification, it trains a classifier using images and class labels only and does not utilize a richer approach to annotations. Both of these tasks are related to scene

categorization. However, in our scene categorization task, we want to learn what *type* of scene an image is (e.g. *bedroom*, *mountains*, etc.), rather than assess certain qualities about it as in (8), and we look at the use of annotations with rationales to improve the usefulness of the annotation, rather than looking at improving the image representation as in (9).

2.2 Improving Annotation Effectiveness

Other work has explored ways to improve the effectiveness or efficiency of human annotations for learning object or scene categories from training examples.

Active learning methods survey unlabeled image examples and decide for which of these a label would be the most useful or informative (10, 11, 12, 13, 14, 15). This work could potentially be paired with ours to make more efficient use of annotator time, though we don't explore this possibility here.

Some work has explored the use of **games** to improve the quality of annotations, where the game aspect incentivizes reliability in annotations, such as The ESP Game (16) and Peekaboom (17). In future work, we could reapply such ideas to use games in ensuring high quality rationale annotations as well.

Other work has used **paid annotators**, e.g., via Amazon's Mechanical Turk, to get a large number of labels more quickly (5, 18). This tends to create some quality control challenges.

Relevance feedback in information retrieval tries to pinpoint the desired content for a specific user, and often entails getting their reaction to some candidate responses, whether positive or negative. This mode of user interaction has been explored in the content-based image retrieval community extensively (10, 19).

In contrast to any of these previous attempts to improve annotation effectiveness, our approach results in annotations that we expect will contain useful information beyond a simple class label.

2.3 Natural Language Processing

In natural language processing, there is work showing interactive new ways to solicit input from annotators about documents aside from the usual classification labels. In (20), Raghavan, Madani, and Jones ask annotators whether a word is relevant or not for a given topic. Druck, Settles, and McCallum look at labeling *features* (words) rather than entire instances in (21). Finally, Zaidan,

Eisner, and Piatko explore the idea of specifying which phrases in a movie review most influenced sentiment classification (positive review vs. negative review) in (1). The method of (1) in particular inspires our approach, as we adapt the authors' idea to the visual domain.

3

Approach

In order to create a classifier that gains insight from annotator rationales, we modify somewhat the traditional approaches to image representation and SVM classification.

3.1 Image Representation

In order to perform image classification, we first need to come up with a representation of the images on which we will train and test our classifier. We would like our representation to be localizable, so that it is possible to remove features only from certain portions of the image to form a contrast example (as explained in later sections).

Our approach to image representation for both the Fifteen Scene Categories and Hot or Not data sets is typical of recent computer vision research. We first find SIFT descriptors for each image in the set. We then randomly select a number of the resulting SIFT descriptors over the entire corpus of images, and cluster them using k-means, giving us a bag of k words. From the k centers found, we map each SIFT descriptor in an image to the index of the center nearest to the descriptor (minimum Euclidean distance) found by k-means. For each image, we create a bag of words histogram $(b_1 \ b_2 \ \dots \ b_k)$, where any b_i in the histogram is the number of SIFT descriptors in the image with nearest center i .

These histograms taken over an entire image, henceforth referred to as “original examples,” will serve as a subset of the training examples that we use to train a support vector machine (SVM), after vector normalization. The remainder of the training examples (“rationale examples”) will be

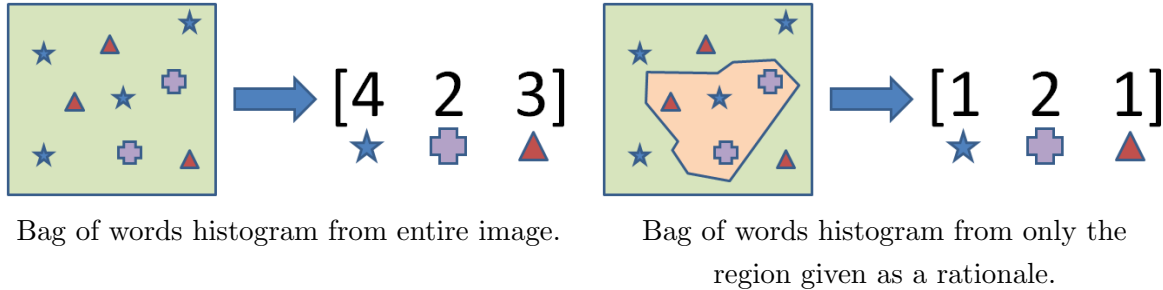


Figure 3.1: Bag of Words Histogram Example - We find significant features in the image then map them to their closest corresponding mean found by k-means ($k = 3$ in this example). These features are represented by the stars, pluses, and triangles. Then, we transform all the significant features in the image into a bag of words histogram (left). We also perform this same transformation on the part of the image inside a rationale polygon (right).

based on bag of words histograms taken only over the regions inside of a rationale polygon. See figure 3.1 for example. With these training examples, we train a linear SVM.

In the following sections, details of our representation for each data set are provided.

3.1.1 Scene Categories Data

To form a representation of the scene categories data, we first run Lowe’s SIFT keypoint detector (22) on each image in the data set. This keypoint detector does not use dense descriptors; it takes descriptors only at points of interest, which are determined using an edge detector (difference-of-Gaussians or DoG) run at several different scales (see figure 3.2 for comparison). We take a subset of around 200,000 of the resulting SIFT descriptors, and use k-means as described above (with $k = 500$) to give us a frequency vector for each image, giving us the original examples for this data set.

3.1.2 Hot or Not Data

Our representation of the Hot or Not data is similar to that of the scene categories data, with some minor differences. Rather than using Lowe’s SIFT keypoint detector (22), we collect dense descriptors every two pixels at a single scale of eight pixels, using the VLFeat library (23). We chose to use dense features for this data set because we are interested in ensuring that we capture a large number of features in the subject’s face and other body parts, and DoG features provide



Figure 3.2: DoG vs. Dense Features - An example of the difference between the points selected on an image (left) by difference-of-Gaussians or DoG (middle) and dense feature selection (right). We use dense feature selection on our Hot or Not data to ensure that a significant number of features are selected from all parts of the person in the image.

no such guarantee (see figure 3.2 for comparison). We again cluster a subset of the resulting SIFT descriptors using k-means ($k = 500$, as before). Finally, we create a frequency vector in the same way as above, and use these frequency vectors as the original examples for this data set.

3.2 Training an SVM with Rationale Examples

At this point, we have two frequency vectors for each image - one original example and one rationale example. We now need a way to incorporate both of these types of training examples into a support vector machine classifier. We might be tempted to simply throw both of these into our training set as they are, but because we will be classifying not rationale examples, but original examples (i.e., full images), we must somehow modify a rationale example in order to claim legitimately that it is a member of the class its image was labeled as. We use the method of (1) to accomplish this, outlined in the following paragraphs.

The intuition Zaidan et al. give in (1) for adding training examples taken only over the rationales (but with a rationale in the case of (1) being a sequence of words in a movie review text, rather than a polygon in an image) is as follows: for any original example \vec{x}_i with a rationale polygon \vec{r}_i (or the union of all rationale polygons if there are multiple) in its image, we can create a “contrast example” \vec{v}_i by masking out the area of the image inside this rationale (i.e., taking only the descriptors that fall outside of the rationale polygon), and the SVM should thus not be as

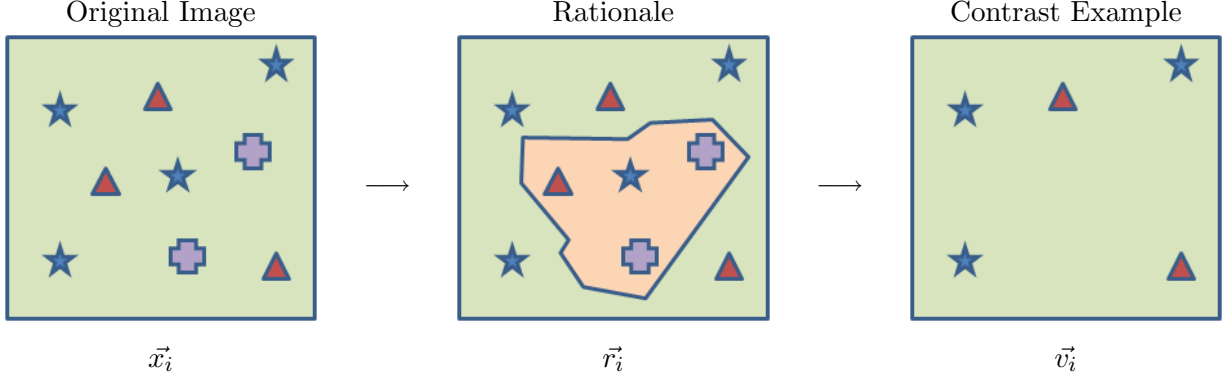


Figure 3.3: Original, Rationale, and Contrast Examples - \vec{x}_i is our original example (left), \vec{r}_i is the rationale annotation (middle), and \vec{v}_i is the contrast example (right), which is the original example with the rationale masked out. We should be less confident in the class label y_i for the contrast example due to its lack of features that the annotator found significant in his or her class choice.

confident in its classification of this contrast example due to the fact that this potentially important region was masked out. (See figure 3.3 for example.) Hence, we ask the SVM to not only maximize the margin between positive and negative examples, but also maximize the margin between positive examples and positive contrast examples by finding \vec{w} such that $\vec{w} \cdot \vec{x}_i - \vec{w} \cdot \vec{v}_i \geq \mu$, where μ is the desired size of the margin between training examples taken over the full image and those taken over only the rationales (see figure 3.4).

Normally, a soft-margin SVM finds \vec{w} and $\vec{\xi}$ such that

$$\frac{1}{2} \|\vec{w}\|^2 + C \left(\sum_i \xi_i \right) \quad (3.1)$$

is minimized, subject to the constraints

$$(\forall i) \vec{w} \cdot \vec{x}_i \cdot y_i \geq 1 - \xi_i \quad (3.2)$$

$$(\forall i) \xi_i \geq 0 \quad (3.3)$$

where $y_i \in \{-1, 1\}$, representing the true label for training example \vec{x}_i , and ξ_i is a slack variable allowing for a misclassification of x_i . The $C > 0$ parameter controls the cost of such a misclassification.

3.2 Training an SVM with Rationale Examples

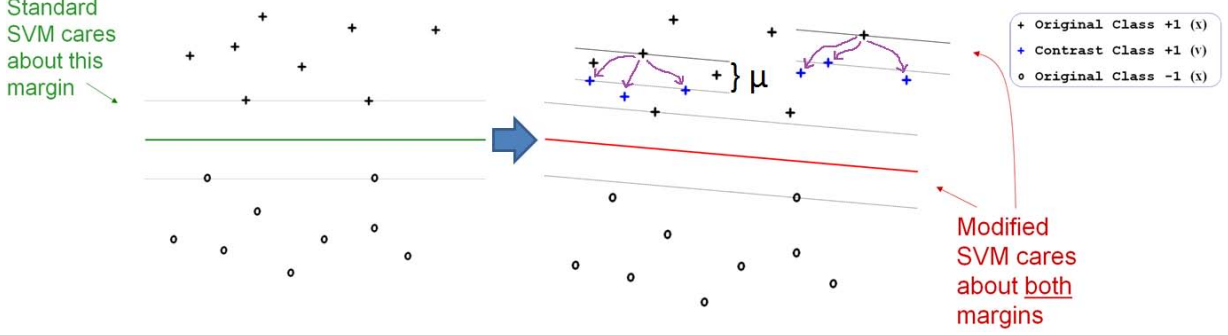


Figure 3.4: SVM Modified for Contrast Examples - We use an SVM to optimize both the margin between classes and the margin between original and contrast examples. The SVM should not be as confident in its classification for a contrast example, as it lacks information that the annotator found important in choosing a class label. This is why we want a margin between positive original examples and positive contrast examples, as well as a margin between positive examples and negative examples. Figure taken from (1).

To allow for our contrast examples, Zaidan et al. (1) suggest adding the constraint:

$$(\forall i) \vec{w} \cdot (\vec{x}_i - \vec{v}_i) \cdot y_i \geq \mu(1 - \eta_i) \quad (3.4)$$

where \vec{v}_i is one of the contrast examples we constructed from example \vec{x}_i , η_i is the corresponding slack variable, and μ is the soft-margin for our contrast constraints. Now, we ask the SVM to find \vec{w} , $\vec{\xi}$, and $\vec{\eta}$ to minimize

$$\frac{1}{2} \|\vec{w}\|^2 + C \left(\sum_i \xi_i \right) + C_{contrast} \left(\sum_i \eta_i \right) \quad (3.5)$$

where $C_{contrast}$ is the cost of a misclassification of a contrast example.

Finally, we divide the constraint in 3.4 by μ in order to get a rationale example \vec{r}_i with a constraint in the form of 3.2, which gives

$$(\forall i) \vec{w} \cdot \vec{r}_i \cdot y_i \geq 1 - \eta_i \quad (3.6)$$

where \vec{r}_i is our new training example (a rationale example), which is defined as:

$$\vec{r}_i = \frac{\vec{x}_i - \vec{v}_i}{\mu} \quad (3.7)$$

To normalize, we divide both the original example \vec{x}_i and the rationale example \vec{r}_i by the magnitude of the original example $\|\vec{x}_i\|$, like Zaidan et al.'s method in (1).

After normalization, this training example \vec{r}_i is added to the normal training set with classification y_i (i.e., using the same classification as the original example from which it was derived), but is given misclassification penalty $C_{contrast}$ rather than the misclassification penalty of C given to original examples. This gives us three scalar parameters per SVM, $(\mu, C, C_{contrast})$, which we set via cross-validation on held-out data.

Finally, in order to allow for a bias term in the hyperplane, we prepend a 1 to each original training example \vec{x}_i and each contrast example \vec{v}_i . Because we take the difference of these to get the rationale example \vec{r}_i , we prepend a 0 to each \vec{r}_i (1).

Our final training set consists of a set of original examples \vec{x}_i and a corresponding set of rationale examples \vec{r}_i .

See figure 3.5 for a summary of our algorithm.

3.3 Issues with the Rationale-Based Approach

There are several potential issues with using rationales to enhance the performance of some classification tasks, a few of which we discuss here.

3.3.1 Amenability to Different Tasks

We believe that this approach should give the greatest performance boost when the data set in question has images that contain a somewhat sparse set of regions that would lead it to be placed in one category rather than another, like many images in the scene categories data set. The task should also be one that is subjective or perceptual, as the reasoning behind a human’s classification decision may be the most ambiguous and complex to learn in such kinds of tasks, thereby creating the greatest opportunity for rationale polygons to steer the classifier in the right direction. For example, in the scene categories data set, an image of a bedroom might have a bed, a dresser, and a mirror to distinguish it from, say, a kitchen; but in the case of either a kitchen or a bedroom, much of the image will likely consist of floor, wall, and ceiling; which are of ostensibly limited use to a human in distinguishing between the two.

Intuitively, there are questions for which this approach, at least in its current form, does not seem appropriate at all. For example, if the objective is to label images as either containing or not containing a vehicle, the idea of giving a rationale, to some extent, falls apart. While it seems

Algorithm Summary - Training a Visual Classifier with Rationales	
1	Find local features (we use SIFT descriptors, with sampling technique dependent on the data set) for all images in data set.
2	Collect human rationale annotations for n of the images in the data set, where n is the maximum desired training set size per class. Each of these annotations contains a class label and one or more polygons around the region(s) most influential in the annotator's class decision.
3	Randomly select a large subset of the features from all images in the data set (on the order of 100,000 features total).
4	Create a bag of words $(\vec{w}_1 \ \vec{w}_2 \ \dots \ \vec{w}_k)$, by clustering the feature subset using k-means (we use $k = 500$).
5	For each image, construct a bag of words histogram $\vec{x}_i = (x_{i_1} \ x_{i_2} \ \dots \ x_{i_k})$ where $(\forall j) x_{i_j}$ is the number of descriptors \vec{d} in the image such that $(\forall l) \text{dist}(\vec{w}_j, \vec{d}) \leq \text{dist}(\vec{w}_l, \vec{d})$ (i.e., \vec{w}_j is the closest word in the bag to \vec{d}). This bag of words histogram is an "original example."
6	Construct a second bag of words histogram $\vec{r}_i = (r_{i_1} \ r_{i_2} \ \dots \ r_{i_k})$ again for each image, just like in (5), but using only those descriptors \vec{d} falling inside of a rationale polygon for the image. This bag of words histogram is a "rationale example."
7	To get our final training examples for image i , we normalize both \vec{x}_i and \vec{r}_i by $\ \vec{x}_i\ $, divide the rationale example by the width μ of the margin between original and contrast examples, and prepend a 1 to original example \vec{x}_i and a 0 to rationale example \vec{r}_i , giving us our final training examples, $\vec{x}_i' = (1, \frac{\vec{x}_i}{\ \vec{x}_i\ })$ and $\vec{r}_i' = (0, \frac{\vec{r}_i}{\ \vec{x}_i\ \cdot \mu})$.
8	Choose a subset of the images from each of the two classes to be learned, and train a support vector machine using both the original examples \vec{x}_i' and rationale examples \vec{r}_i' in the training set for each of these images, with the same class label for each type of example on a given image. For original examples (of either class), use misclassification penalty C . For rationale examples (of either class), use misclassification penalty $C_{contrast}$. The parameters μ , C , and $C_{contrast}$ are set via cross-validation on held out data.

Figure 3.5: Algorithm Summary - A summary of the steps of the algorithm that we use to classify images using our rationale-based approach.

reasonable to draw a polygon around the vehicle in a positive example, the “rationale” for a negative example would simply be the entirety of the image, as one would have to look at every part of the image in order to determine absence of a vehicle.



Figure 3.6: Difficult Rationale Example - The correct label for this image is “coast”, but giving a useful rationale for it is difficult, since no individual region of the image is much more revealing of its coastal quality than any other.

Indeed, even in the scene categories data set itself, there are categories in which many of the images are difficult to annotate with a useful rationale. For example, in the Coast category, some images are simply of beach next to water (figure 3.6). An annotator with the best of intentions might approach this type of image in one of two ways: by drawing a rectangle around the entire image as the rationale, or by drawing a tiny polygon around the curve where the water meets the beach. In the former case, treating it like any other rationale seems to introduce unwanted bias into the SVM, as it results in two training examples (that of the full image and that of the polygon) that are nearly identical (or different by a factor of μ). In the latter case, the tiny polygon might capture very few or perhaps even zero interest points, which is not very helpful to the SVM either. Partly due to this problem, we filter out rationales that contain almost all or almost none of the image.

3.3.2 Selection of Useful Rationales

An image classification system generally will not use the same logic to assign a class label to an image that a human would. For example, an SVM that was trained based on a local feature representation such as SIFT descriptors might make many of its class decisions largely based on repeated textures in an image, such as the tile commonly used for kitchen flooring vs. carpeting in a bedroom, rather than objects, such as a refrigerator in a kitchen vs. a bed in a bedroom, that a human might be more likely to look to for a decision. This raises the question of whether the visual cues a human picked up on will even be relevant to the processes used to classify images using currently known image classification techniques. We believe that the answer will be “yes” in some cases, and “no” in others. To account for the cases in which the human’s rationales will not be very helpful, we train our classification system based not only on the regions of the image that fall inside a rationale

3.3 Issues with the Rationale-Based Approach

polygon, but also the entirety of the image, hence the inclusion of both the original and rationale examples in the final set of SVM training examples. This approach helps fix cases where annotated regions were irrelevant, and the amount of performance lost from regions where this information was not only irrelevant but was actually detrimental will hopefully be outweighed by the performance gains from cases where it is indeed relevant.

4

Data

We explore the utility of our new approach to supervised learning of image classes in the domain of two separate data sets: the Scene Categories data set and a new data set from the once popular Hot or Not website.

4.1 Scene Categories

We chose to use the famous Fifteen Scene Categories data set (24) due to the perceptive nature of classifying an image by its scene type, and the vast number of isolated objects in many of the images. For example, an image of a bedroom might contain several objects that help us classify it as such, e.g., a bed, a dresser, an alarm clock etc.; however, it may also contain plenty of distracting information, such as a lamp or a television, that frequently appear in other scene categories and are thus less useful in helping us classify the image. The presence of a large number of isolated objects, some of which are more useful and others of which are less useful, make the scene categories data set a good candidate for our approach to rationales. This data set consists of 200-400 black and white images from each of fifteen types of indoor and outdoor scenes. See example images in each scene category in figure 4.1.

4.2 Hot or Not

We also created a new data set, using images from <http://www.hotornot.com/>. Hot or Not is a website that shows a visitor an image of a random user (which can be restricted to a certain gender



Figure 4.1: Fifteen Scene Categories - Sample Images - Sample images from each category of the Fifteen Scene Categories data set (24). Rationales can be used to isolate aspects of these images that are most unique to their particular classes, such as the part of a Bedroom image with a bed, or the part of a Kitchen image with a refrigerator.

	Male	Female
Mean	9.088	8.495
Standard Deviation	0.607	1.132
Minimum	6.5	2.8
25th Percentile	8.7	7.7
Median	9.3	8.8
75th Percentile	9.5	9.4
Maximum	9.9	9.9

Table 4.1: Hot or Not Data Statistics - Sample statistics for the ratings of the 1000 men and 1000 women in our Hot or Not data set.

and/or age range) and asks the user to rate the attractiveness of the person in the image on a scale of 1-10 (see figure 4.2 for preview of interface). After the visitor rates the person in the image, he or she can view the average rating over all visitors (rounded to the nearest tenth) and the number of visitors who have rated the image in total.

We have collected 1000 images of males and 1000 images of females from this website, along with their ratings, the number of users who rated the image, the URL, and the short “introduction” that the person in the picture supplied. We only used in our data set images that had been reportedly rated at least 100 times.

After collecting these images, we found some interesting statistics about the data. In general, the ratings seemed quite high, with a mean rating of about 9.1 for men and 8.5 for women. In fact, the *lowest* rating of any man in our data set was a 6.5. See table 4.1 for more statistics.

Despite the relatively small portion of the rating scale that is apparently being utilized, the relative ratings seemed anecdotally accurate; i.e., if one person had a significantly higher rating than another person of the same gender, the one with the higher rating generally seemed more attractive than the one with the lower rating.

This data set is an excellent fit for testing our approach in terms of our intuition that rationales are essential to learn a subjective classification task well, and this is about as subjective as tasks come (“beauty is in the eye of the beholder”). Rather than simply discarding all information prior to the final conclusion, the inclusion of rationales in our annotations has the potential to give our classifier much richer insight into the thought process of the human annotator.

The screenshot displays the Hot or Not website interface. At the top, a yellow banner prompts the user to "Select a rating to see the next picture." with a scale from 1 to 10, ranging from "NOT" to "HOT". Below this, there are filters for "Show me" (set to "men and women") and "Age" (set to "any").

On the left, a yellow box shows the "Official Rating" of 9.5, based on 34622 votes, for a user named Jade. Below this is a small photo of Jade and links to "Add to Favorites" and "Meet Me". A yellow button at the bottom left says "Upload a Photo Now!".

The main area features a large video of a woman with long blonde hair, wearing a black top, in a kitchen setting. Below the video, there is a "Share Link" field with the URL <http://www.hotornot.com/r/?eid=SQBQE8R-LN>, and a "Share On:" section with buttons for Facebook, My Space, Bebo, Twitter, and "Share by email".

At the bottom, there is a "Nominate for 'Best Of' | Flag" link and a "Her Introduction" section where Jade writes: "Hi im Jade, im only on here for fun, and to meet friends. im in the collusion fan club. :)"

Figure 4.2: Hot or Not Interface - An example of the interface found on the Hot or Not website, <http://www.hotornot.com/>. Includes rating buttons 1-10 at the top, the image of the person to be rated below (with their introduction underneath the image), and, on the left, the rating and number of votes for the previously rated image.

5

Annotations

Central to our approach are the human annotators, who will provide rationales for each image that we hope will give deep insight into the reasoning that went into their class decision. We use Amazon’s Mechanical Turk (MTurk)¹ to gather most of our annotations in order to create a large data set with a wide variety of annotation styles. Gathering annotations from a large number of sources fits well with our belief in rationales as an approach to subjective tasks.

5.1 Scene Categories

We crowdsourced our annotations of the Fifteen Scene Categories data set (24) to MTurk. In general, the quality of the results varied greatly. In addition to written instructions for annotating an image, our instructions included three sample images from each category (which were removed from the remainder of this study), and a demo video showing how a single annotation is done using the interface. We imposed just two absolute requirements on our annotators (i.e., work that failed to adhere to either of these instructions would be automatically rejected without pay): that they draw at least one polygon per image (which must consist of at least three vertices, by definition), and that they must click the same class label for each polygon they draw. For more details and interesting statistics on the annotations, see table 5.1.

Our goal was originally to have three annotators take on each image, but due to the overwhelming lack of good responses, we repeatedly posted poorly done annotations until we felt we had an

¹MTurk (<https://www.mturk.com>) allows requesters to post small tasks, called “HITS” (Human Intelligence Tasks), to its website for users, called “workers,” to complete for a configurable amount of pay per task.

Annotation Task Summary	
Jobs Posted	34,021
Accepted	8055 (23.7%)
Rejected	25,966 (76.3%)
due to No True Polygons	25,901 (99.7%)
due to Multiple Classes	118 (0.5%)
# Unique Workers	545
Mean Jobs/Worker	62
Total Man-Hours	205
Mean Time/Job	21.7 seconds
Total Man-Hours on Approved Jobs	102
Mean Time/Approved Job	45.4 seconds
Correct Class Label	26,497 (77.9%)
Incorrect Class Label	7329 (21.5%)
Multiple Class Labels	118 (0.3%)
No Class Label	77 (0.2%)
0 Polygons	25,901 (76.1%)
1 Polygon	7490 (22.0%)
2 Polygons	400 (1.2%)
3 Polygons	161 (0.5%)
4 Polygons	41 (0.1%)
5 Polygons	16
6 Polygons	7
7 Polygons	4
8 Polygons	1
9+ Polygons	0

Table 5.1: Scene Categories MTurk Run Statistics - A set of summary statistics from our scene categories annotation run on Amazon’s Mechanical Turk.

adequate data set size, which came out to an average of about 1.80 annotations for each image in the data set (other than those used as examples of the class in the instructions).

Overall, we found the majority of the results in which workers followed the instructions to be quite good (figure 5.1). Some workers drew very tight polygons around objects of interest (figure 5.2), which we explicitly noted was not mandatory in our instructions. A small yet vocal minority took a bit of artistic license with their annotations (figure 5.3). In spite of the likely detrimental effect these “artistic” annotations had on the performance of our approach, we allowed these rationales to be used in our data set on the philosophical grounds that the rationales are subjective. We also left all incorrect labels intact, training based on the label the annotator provided, whether correct or not (but still testing only on ground truth). The only results we pruned from our data set were those that (a) had no polygons marked, (b) had different class labels associated with different polygons, (c) captured almost all ($> 95\%$) of the image inside a polygon, (d) captured almost none ($< 5\%$) of the image inside a polygon, or (e) had one or more “polygons” with edges crossing (figure 5.4).



Figure 5.1: Examples of normal annotations.



Figure 5.2: Examples of annotations with especially tight bounds.

There were several categories in which it appears that some images were non-trivial for even the human annotators to correctly label (see table 5.2). Unsurprisingly, there was a high rate of human confusion for closely-related category pairs such as inside city vs. tall building, inside city vs. street, street vs. highway, and open country vs. mountain. This human confusion likely impacted



Figure 5.3: Examples of “artistic” annotations.



Figure 5.4: Pruning Examples - Examples of each different pruning case, (a) no polygons, (b) multiple class labels (Tall Building and Highway in this case), (c) almost entire image, (d) very little of image, (e) crossed edges. (Best viewed in color.)

our SVM’s power to correctly classify these ambiguous categories, but we would like to think that incorrect category labels is another way in which rationales can be valuable. For example, there were many images in the Open Country category that had mountains in their background, so if an annotator had incorrectly labeled such an image as Mountain but used as his or her rationale the mountains that were indeed in the background, it is still possible that we gain something from this annotation. Had the annotator not given a rationale, however, this example would likely only hurt our results.

5.2 Hot or Not

With the Hot or Not data, we have both annotated a small subset of the images ourselves (at least 100 images in each of the “Hot” men, Hot women, “Not” men, and Not women classes) and crowdsourced a larger portion of the images to Mechanical Turk. For this data set, because we wanted to use the scores from the Hot or Not website as our classifications since they were the average of ratings taken from hundreds of people (a more robust estimate of “groundtruth,” if such a thing exists for this task, than a single individual’s opinion), we did not ask MTurk annotators to give a “Hot” or “Not” class label to the image presented in the MTurk job. We instead told

Human Confusion Matrix (%)																
	NL	B	C	F	H	ID	IC	K	LR	M	OF	OC	SO	SR	SU	TB
B	0	96	0	-	-	-	-	-	3	-	-	-	0	-	-	-
C	0	-	87	2	0	0	1	-	-	4	-	4	-	0	2	0
F	0	0	1	83	0	-	0	-	-	6	0	7	-	1	2	0
H	0	0	0	0	84	1	3	0	-	1	0	2	0	7	1	0
ID	0	-	1	0	0	89	1	0	0	0	0	1	1	1	1	4
IC	0	0	0	0	0	3	26	0	1	0	5	1	7	15	17	25
K	0	0	-	-	-	0	-	88	8	-	2	0	1	-	0	0
LR	0	2	-	-	-	0	0	1	96	-	1	-	0	-	0	0
M	0	0	1	2	0	-	-	-	0	95	-	2	-	-	0	0
OF	0	1	-	0	-	0	-	0	9	-	88	-	2	0	-	0
OC	0	-	7	17	1	0	0	-	-	25	-	43	-	1	5	-
SO	-	0	-	-	0	1	2	2	1	0	2	0	90	1	0	0
SR	0	-	0	-	4	0	24	-	-	1	0	1	0	61	2	6
SU	1	0	1	1	-	0	5	-	6	-	1	3	0	3	75	4
TB	0	-	1	-	-	1	11	0	-	0	0	1	-	1	1	85

Confusion Matrix of our “Original Examples Only” Baseline (%)																
	NL	B	C	F	H	ID	IC	K	LR	M	OF	OC	SO	SR	SU	TB
B	-	38	2	1	4	4	2	13	10	2	6	0	5	5	2	6
C	-	1	64	4	9	0	0	1	0	12	0	6	0	0	2	1
F	-	0	0	91	0	0	0	0	0	4	0	1	2	0	2	0
H	-	2	12	0	67	5	1	0	0	1	0	1	1	4	3	3
ID	-	10	4	0	8	18	11	6	4	1	4	0	16	11	5	2
IC	-	3	1	1	7	3	27	18	3	1	5	0	8	7	1	15
K	-	13	1	0	2	6	9	28	17	0	15	0	4	3	2	1
LR	-	15	0	0	2	5	4	11	31	1	14	0	7	1	8	2
M	-	2	9	11	6	0	0	0	0	62	0	3	1	1	4	0
OF	-	9	0	0	1	3	6	15	18	0	45	0	1	1	1	0
OC	-	2	21	11	6	1	0	0	0	22	0	32	0	0	2	1
SO	-	3	0	3	0	5	4	3	6	1	2	0	62	6	4	1
SR	-	1	1	1	3	13	15	1	4	2	0	0	14	34	6	7
SU	-	1	0	1	1	1	0	1	3	2	2	0	2	0	85	2
TB	-	2	15	1	5	6	7	3	2	1	3	2	2	1	2	48

Table 5.2: Human Confusion Matrix for Scene Categories - The confusion matrix for our run of the scene categories data set on MTurk (top) compared with the confusion matrix for our baseline classifier (bottom). Some categories are difficult even for humans to classify correctly, such as Inside City (frequently labeled as Tall Building, Suburb, or Street), where our baseline classifier’s performance was higher than human performance. Labels across top are No Label (annotator didn’t choose a class label), Bedroom, Coast, Forest, Highway, Industrial, Inside City, Kitchen, Living Room, Mountain, Office, Open Country, Store, Street, Suburb, Tall Building.

the MTurk worker that the consensus among at least 100 people was that the person in this image is attractive or unattractive, and instructed them to draw polygons around the parts of the image they think best demonstrates the attractiveness or unattractiveness, as decided by the Hot or Not rating. In hopes that it would improve the quality of annotations, we also asked that the worker add a comment on each polygon he or she annotated indicating the thought behind the choice, but did not enforce this requirement for the purpose of deciding whether to pay the worker for the job. See figure 5.5 for an example of our MTurk interface for this task. Many workers did adhere to this requirement, adding labels to their polygons such as [sic] *CUTE FACE*, *soldier*, *girl*, *fat*, *athletic upper body*, *THICK EYEBROW*, *pimple*, and *long beautiful hair*. As in the scene categories MTurk run, the instructions included a video example of the way we wanted workers to do their annotations, with annotations shown for four images (one from each class) which we removed from the dataset.

In general, the quality of these annotations was quite high (see figure 5.6 for samples). In contrast with the scene categories run, the vast majority of workers annotated at least one polygon.¹

See table 5.3 for statistics on the Mechanical Turk run of the Hot or Not data.

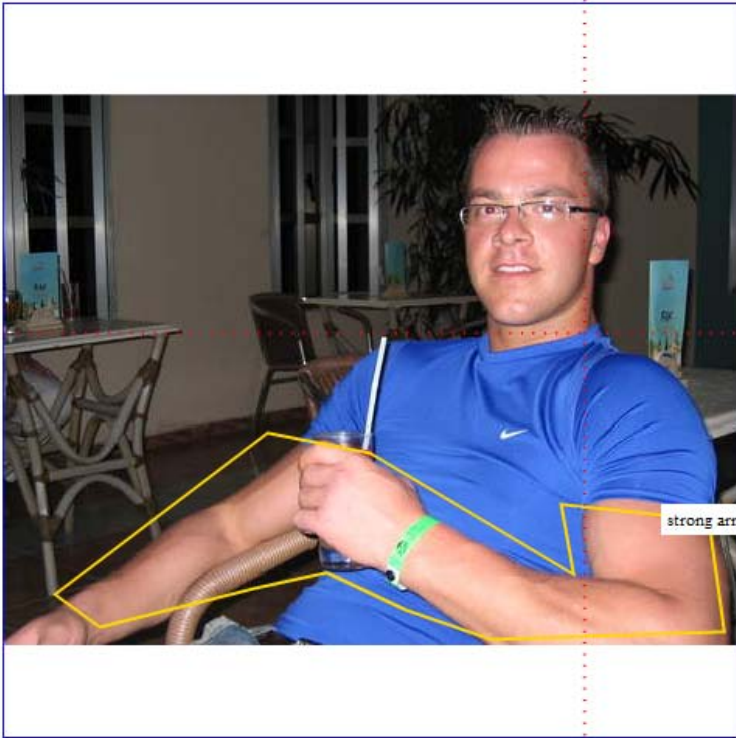
¹This was perhaps due to the fact that in this case, there was no requirement to choose a class label, so had the worker not drawn a polygon he or she would have done no work for the job, making the requirement to do something beyond giving a class label potentially less ambiguous.

Outline at least one polygon (with relevant comment) per image you annotate. Failure to follow this instruction will result in your annotations being rejected automatically.

The consensus among at least 100 people was that the man in this image is **attractive**. Please outline the region(s) of this image which you find most ATTRACTIVE about this man. This could be (for example) his entire body, his face only, his torso only, or a certain portion or portions of each.

Click the button labeled "attractive" or "unattractive" to the right of the image to begin drawing a polygon, then click points around the region of interest to draw a polygon. Select the "type-object-name" text to enter a comment, then click "Done". Repeat to draw additional polygons.

When you are finished, press the "Submit results" button below the image.



Object:

type-object-name

Done

Cancel

Undo

strong arm

Image by: (loading) License: loading

Submit results

Figure 5.5: Hot or Not MTurk Interface - An example of the interface we used to gather rationale annotations on Mechanical Turk for our Hot or Not data set. The worker is informed in the instructions that the man in the image was considered attractive based on the rating we found for him on the Hot or Not website. In this case, the annotator's rationale for this man's attractiveness is a polygon around his arms labeled with "strong arms."

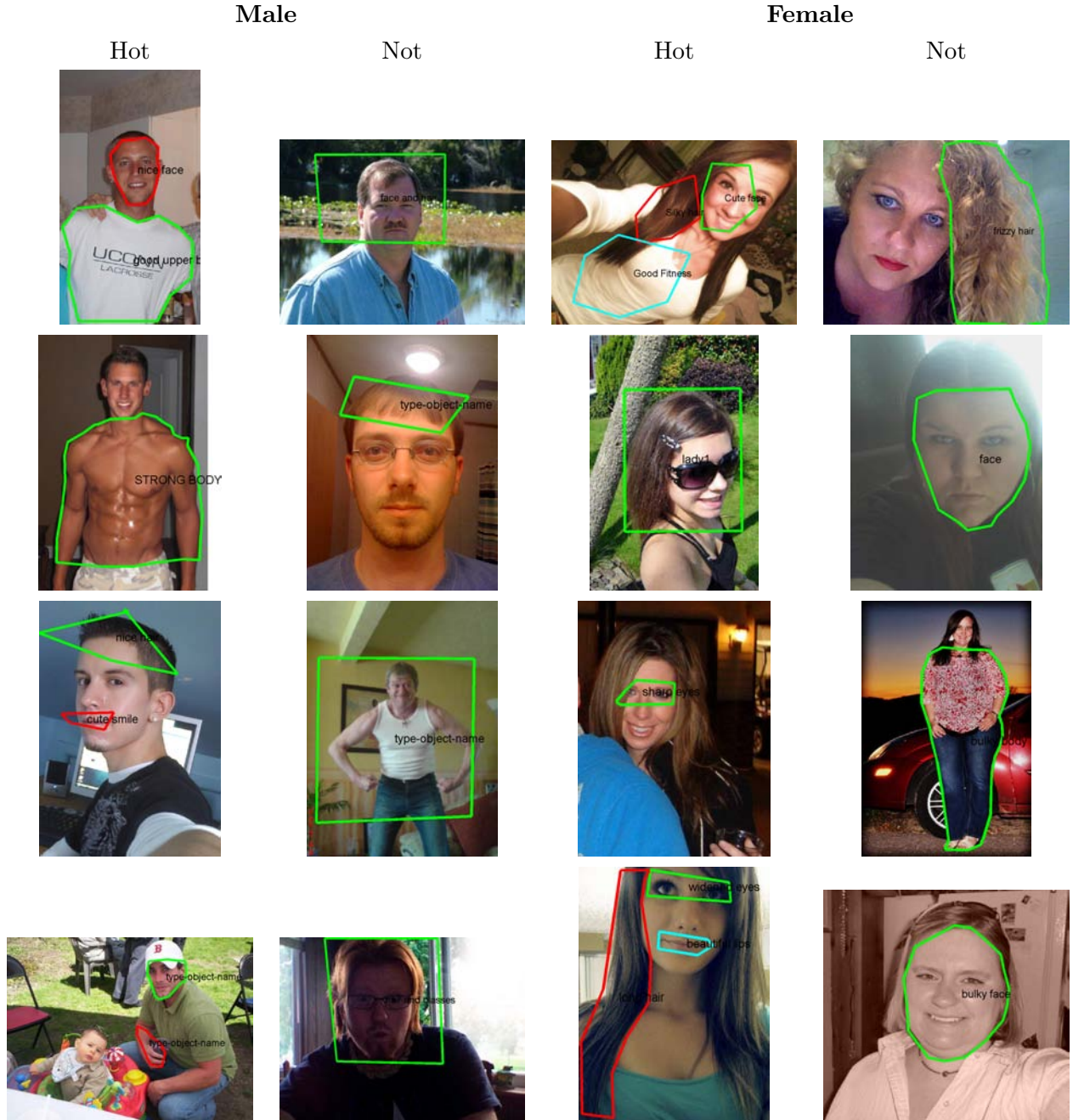


Figure 5.6: Hot or Not Annotations from MTurk - For the Hot or Not task, annotators were shown an image of a man or a woman rated in either the top or bottom 25% in our data set, and were asked to outline what they found attractive about the person in the image if it was from the top 25%, or what they found unattractive about the person in the image if it was from the bottom 25%. They were also asked to type a short label for each polygon. Generally, most of the annotators seemed to take the task seriously and produced high quality rationale annotations.

Annotation Task Summary	
Jobs Posted	2000
Accepted	1845 (92.3%)
Rejected (due to No True Polygons)	155 (7.8%)
# Unique Workers	104
Mean Jobs/Worker	19
Total Man-Hours	43
Mean Time/Job	77.5 seconds
Total Man-Hours on Approved Jobs	42
Mean Time/Approved Job	82.4 seconds
0 Polygons	155 (7.8%)
1 Polygon	1660 (83.0%)
2 Polygons	162 (8.1%)
3 Polygons	19 (1.0%)
4 Polygons	3 (0.2%)
5 Polygons	0
6 Polygons	1 (0.1%)
7+ Polygons	0

Table 5.3: Hot or Not MTurk Run Statistics - A set of summary statistics from our Hot or Not annotation run on Amazon’s Mechanical Turk.

6

Results

We would like to show that annotator rationales can be of value to building classifiers for different tasks. In attempting this, for each task, we will train a classifier using our approach (a training set consisting of both original examples and rationale examples), and using the traditional approach (original examples only) as a baseline, in addition to another baseline where we use only the rationale examples as our training set. We will also add a few extra baselines for each of the two tasks. For the scene categories task, we add a mutual information baseline, which automates discriminative feature selection, as described by Dorko and Schmid in (25). For the Hot or Not task, we add facial recognition baselines.

To evaluate results from the scene categories task, we perform several trials by training on a fixed set (across our approach and baselines) of randomly selected training examples and then test on the rest of the data, calculating precision-recall curves and mean average precision per class. We then compare these metrics on our approach to the same metrics for each of our baselines.

To evaluate performance of the Hot or Not task, we again perform trials by training on a fixed set of randomly selected training examples, and then test on the rest of the data, calculating the percent of test examples correctly classified.

6.1 Scene Categories

We begin by evaluating our approach on the Scene Categories data set.

6.1.1 Methodology

We have tested our approach on our Mechanical Turk annotation data from the Fifteen Scene Categories data set using the aforementioned procedures against three baselines. In one trial run, we train a binary classifier for each of the fifteen classes and measure its precision and recall¹ at different SVM decision thresholds² to find a precision-recall curve for each class. We also compute the mean average precision for each classifier to give a single numerical point of comparison between our approach and the baselines. In each trial, 25 images from each class (that were not held out for parameter optimization) are randomly selected to be used as training examples and the same training set is used for our approach and each of the three baselines, for a training set size of 25 positive examples and $(15-1) \times 25 = 350$ negative examples for each classifier. Then, 100 images per scene category that were not selected as training examples (or held out for parameter optimization) are randomly selected as test examples, for a total of 1500 test examples (100 positive and 1400 negative for any particular classifier).

100 trial runs were performed to get the following results.

6.1.2 Baselines

To evaluate our approach’s performance on the classification task, we compare our performance with that of three baselines.

The first is the **Originals Only** baseline. This baseline can be thought of as our comparison to the typical approach to image classification, as it uses only the original training examples with the full bag of words histograms, making no use of rationales. Beating this baseline would suggest that there is reason to make use of rationales for scene classification over the traditional approach.

Our second baseline is the **Rationales Only** baseline. For this baseline, we *only* use the rationale examples. Beating this baseline would suggest that rationales are not merely foreground

¹In the case of a classifier for the “Bedroom” class, for example, we define “retrieved images” to be those images that our classifier labeled as Bedrooms. “Precision” is then the number of retrieved images that are actually in the Bedroom class divided by the total number of retrieved images – a measure of false positives. “Recall” is the number of retrieved images that are actually bedrooms divided by the total number of bedroom images in the corpus – a measure of false negatives.

²To come up with these thresholds, we rank the test examples by their decision value and take a threshold (and hence get a precision and recall value) at each one of them.

segmentation, as this baseline would essentially be the result of cropping out all parts of the image that don't fall into a rationale polygon.

Our last baseline is the **Mutual Information** baseline. We approach Mutual Information as described by Dorko and Schmid in (25). It is an automated way of selecting the k most discriminative features (in our case, the most discriminative words in our bag of words histograms) between the two classes for which we want a classifier. We set $k = 100$, so this baseline takes the top 100 words from our bag of 500 words. Beating this baseline would suggest that rationales are more powerful than automated feature selection, a necessary condition for rationales to be useful, as we would prefer to use an automated approach over a manual one if they are equally powerful.

In testing each of these baselines, we use the same images in our training and test sets, and perform parameter cross-validation in the same way as our approach.

6.1.3 Results of Scene Categories Experiment

Our test showed an improvement in mean average precision with our approach over the maximum baseline per class in the cases of 13 out of the 15 scene categories (see table 6.1). We also show the precision-recall curve plots for the 8 image categories most improved by our approach in figure 6.1.

6.1.4 Discussion

We discuss the extent to which our approach to using rationales is successful in creating classifiers for scene categories and attempt to explain the differences in success by scene category.

6.1.4.1 Success of our Approach

We have shown that for the majority of scene categories, using annotator rationales can be very helpful in building a binary SVM classifier. The mean average precision of our approach beat all three baselines for thirteen of the scene categories, and the improvement is statistically significant with $\alpha = 0.1$ in eleven of the thirteen.

Having beaten the Originals Only baseline in all but one category suggests that there is value in using rationales over the traditional approach of using only original examples. We've shown that rationales must provide some insight into the differentiation of scene categories that a simple class label cannot.

Mean Average Precision per Scene Class						
#	Class Name	Ours	Originals Only (P-Value)	Rationales Only (P-Value)	Mutual Information (P-Value)	Gain Over Maximum Baseline
1	Kitchen	0.1395	0.1196 (0.0000)	<i>0.1277</i> (0.0000)	0.1202 (0.0000)	+0.0879
2	Living Room	0.1238	0.1142 (0.0000)	0.1131 (0.0000)	<i>0.1159</i> (0.0000)	+0.0656
3	Inside City	0.1487	0.1299 (0.0000)	<i>0.1394</i> (0.0000)	0.1245 (0.0000)	+0.0644
4	Coast	0.4513	<i>0.4243</i> (0.0000)	0.4205 (0.0000)	0.4129 (0.0000)	+0.0617
5	Highway	0.2379	<i>0.2240</i> (0.0000)	0.2221 (0.0000)	0.2112 (0.0000)	+0.0603
6	Bedroom	0.3167	<i>0.3011</i> (0.0621)	0.2611 (0.0000)	0.2927 (0.0055)	+0.0505
7	Street	0.0790	<i>0.0778</i> (0.0000)	0.0766 (0.0000)	0.0775 (0.0000)	+0.0159
8	Open Country	0.0950	0.0926 (0.0000)	<i>0.0946</i> (0.0003)	0.0941 (0.0000)	+0.0036
9	Mountain	0.1158	0.1154 (0.0322)	0.1151 (0.0004)	<i>0.1154</i> (0.0793)	+0.0028
10	Office	0.1052	<i>0.1051</i> (0.0566)	0.1051 (0.0082)	0.1048 (0.0000)	+0.0007
11	Tall Building	0.0689	0.0688 (0.0003)	<i>0.0689</i> (0.0512)	0.0686 (0.0000)	+0.0006
12	Store	0.0867	<i>0.0866</i> (<i>0.3187</i>)	0.0857 (0.0000)	0.0866 (0.1526)	+0.0004
13	Forest	0.4006	0.3956 (0.0000)	<i>0.4004</i> (<i>0.2750</i>)	0.3897 (0.0000)	+0.0003
14	Suburb	0.0735	0.0735 (0.2908)	0.0737 (0.9965)	0.0733 (0.0000)	-0.0027
15	Industrial	0.1046	0.1056 (0.7757)	0.0911 (0.0000)	0.0981 (0.0000)	-0.0099

Table 6.1: Scene Categories Mean Average Precision - A table of the scene categories showing their mean average precisions with rationales (column “Ours”), Originals Only baseline, Rationales Only baseline, Mutual Information baseline, with the P-Value for a one-sided t-test of our improvement over each of the baselines, and the improvement of our approach over the best of the three baselines per class (“Gain Over Maximum Baseline”). Classes are sorted in order of highest gain over the top baseline. In all but two cases (classes Suburb and Industrial), our mean average precision is higher than any of the three baselines, and of the thirteen classes where we beat all three baselines, our mean average precision was *statistically significantly* higher at $\alpha = 0.1$ than any of the other three baselines for all but two classes (Store, Forest). So, our approach had a statistically significant win over any other method for eleven out of the fifteen (11/15) classes.

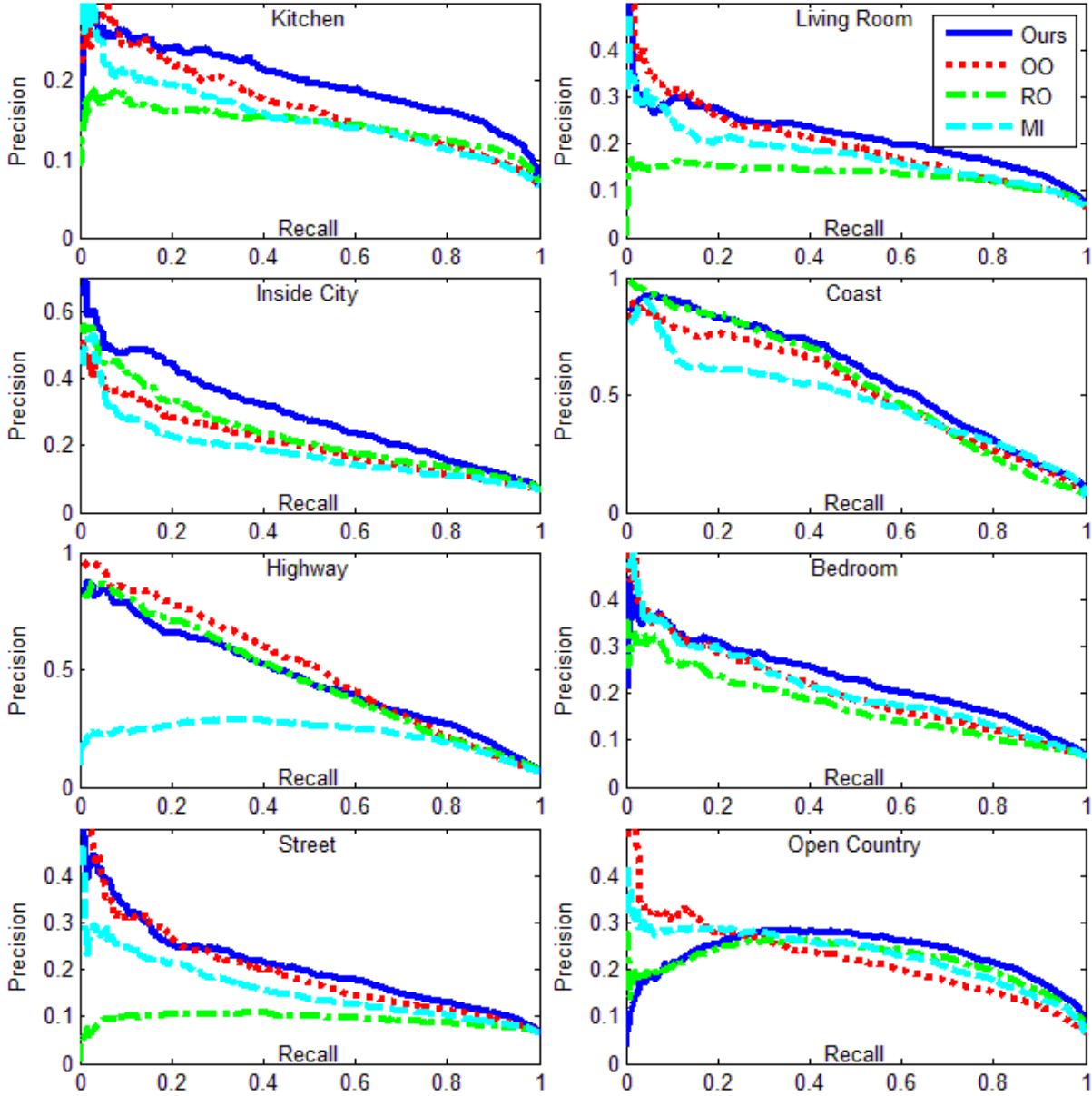


Figure 6.1: Scene Categories Precision-Recall Curves - Precision-recall curves for the eight scene categories most improved by our approach (based on gain in mean average precision of our approach over the top baseline). “Ours” curves represent the performance for our approach, “OO” curves represent the performance of the Originals Only baseline, “RO” curves represent the performance of the Rationales Only baseline, and “MI” curves represent the performance of the Mutual Information baseline.

Beating the Rationales Only baseline in all but one category legitimizes this approach in the sense that we would not get the same results by simply cropping out the parts of the image that do not lie in a rationale polygon. We can claim that, on this task, altering the SVM to support two soft-margins as Zaidan et al. suggest in (1) gives superior performance to simply training on a subset of an image based on its foreground segmentation.

Finally, beating the Mutual Information baseline in every scene category tells us that human insight has value beyond the insight of a simple mathematical formula. Our approach cannot be rejected on the grounds that the same thing could be accomplished more quickly using automated discriminative feature selection.

In summation, our results reveal that using rationales to learn binary scene classifiers is more powerful than the traditional approach to learning a classifier, a foreground segmentation, and automated discriminative feature selection. The human reasoning behind a class selection is more powerful than the class selection alone.

6.1.4.2 Intercategory Performance Disparities

A glance at the gains in mean average precision in table 6.1 and the different shapes of the precision-recall curves as shown in figure 6.1 reveals significant differences in how rationales affect classification performance for a given scene category. Our approach most benefits classification performance of the classes Kitchen, Inside City, and Coast, and least benefits (or hurts) the classes Industrial, Store, and Tall Building. See figure 6.2 for sample images from these categories.

With some observation, these disparities aren't difficult to explain. In the classes most benefited by our method, there is generally a great deal of distracting visual information in the images, from which rationales could help steer away the classifier. In the Kitchen category, for example, we can find many images with ostensibly irrelevant local features, such as rich textures in the wall and floor tiling that might be highly influential in the classifier's choice if it didn't have the rationales to steer it in the direction of more discriminative aspects of the Kitchen category. On the other hand, in the classes least benefited by our method, there is a very high concentration of evidence of the image's category at most of its high gradient points, which are where our descriptors are concentrated. In the Forest category, for example, nearly every high gradient point in the image is part of a tree, a highly representative object of the Forest class. Essentially, for the classes our approach performed the worst on, the DoG interest point detector was already detecting the points that would give the

most evidence that an image is a member of its class, acting as a type of rationale on its own in these categories, and additional rationales provide little benefit in that situation.

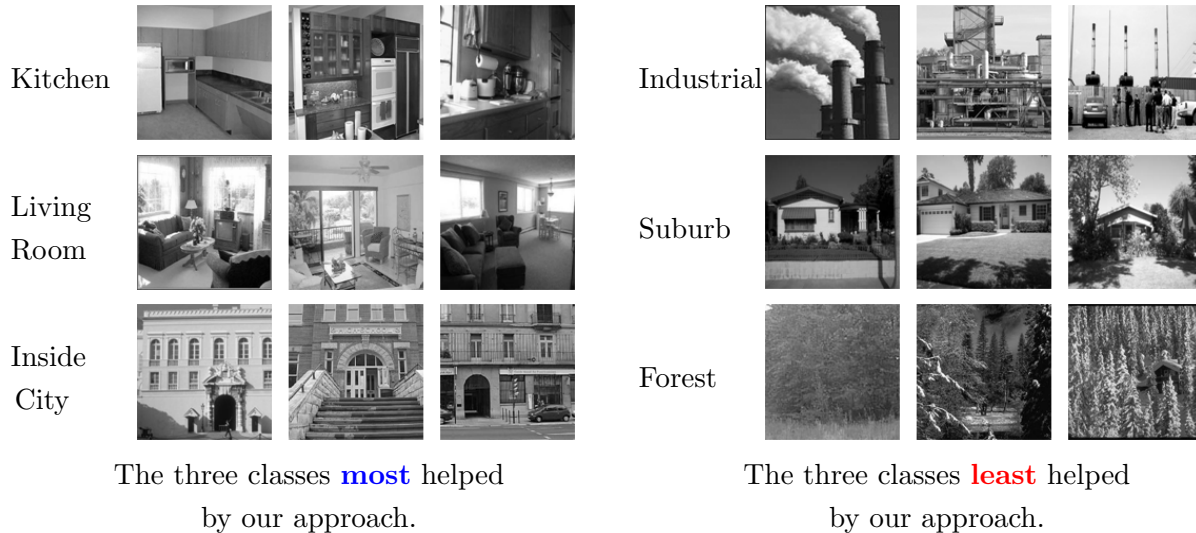


Figure 6.2: Best and Worst Scene Categories - Sample images from the scenes data set of those categories that our approach benefited the most (left) and the least (right).

6.2 Hot or Not

We would like to evaluate the usefulness of rationales for a highly subjective classification task, so we also test our approach on a new data set with images from the Hot or Not website, designing a classifier that can take an image of a human and decide whether he or she is “Hot” or “Not.”

6.2.1 Methodology

We tested our approach on both our own rationale annotations and those gathered from workers on Mechanical Turk. We separated our tests by sex, testing classification ability for men and women separately. In order to create a binary classification task (in particular, “Hot” vs. “Not”) from a set of images that have been rated as real numbers, we must choose a threshold for what should be considered Hot and Not. Because it would be extremely difficult for even a human to tell beyond a random guess whether an image near the median Hotness is actually in the top or bottom half, we chose the more reasonable task of considering the images with top quartile (25%) ratings in our

data set to be considered Hot, and bottom quartile ratings to be considered Not, disregarding the middle 50% entirely for the sake of the experiment. Anecdotally, given that an image is either in the top or bottom quartile of ratings, it is not difficult as a human to guess which of the two it is in the vast majority of cases. This threshold parameter can, of course, be modified to create easier or harder classification tasks, but we will stick with a threshold of 25% as a simple point of comparison between our method and the baseline.

In one trial run for a single sex, we train a binary classifier on N randomly selected images with both rationale examples and original examples (our approach), with rationale examples only, and with original examples only (the traditional approach) from each of the top and bottom quartile, labeling them as “Hot” and “Not”, respectively. Without rationales, this gives us a training set size of $2N$ (N Hot examples, N Not examples). 100 trial runs were performed for each of our training set size parameters: $N = 25, 50, 75, 100$.

6.2.2 Face Detection Baseline

For this data set, we also consider a baseline that focuses the classifier’s attention by using an automatic face detector. This baseline will help show to what extent the Hot or Not class decisions are determined based on the face alone. If it turned out that existing facial detection methods serve as better “rationales” than humans can give, then certainly it would be a waste of the annotator’s time to manually annotate these images. For that reason, we have performed separate¹ trials using the Viola-Jones face detector as a baseline.

We do this in two ways. First, we take the bounding box output by the face detector and use it as if it were a rationale given by a human annotator, including both the original example and the rationale example in the training set just as we do with our approach to rationales, and classifying original images. The second face detection baseline we use is to simply treat the area inside the bounding box as the entire image, throwing the rest away, and using as test examples only the area inside the bounding box as well. In other words, we both train and test on faces only for this second face detection baseline.

¹In the face detection trials, we had to use a different, but still randomly selected training set, due to the lack of faces found by the detector in some of the images for which we had rationales.

Male				
Training Examples per Class	$N = 25$	50	75	100
Ours (Our Annotations)	55.40%	56.96%	58.44%	60.01%
Ours (MTurk Annotations)	53.73%	54.24%	54.58%	54.92%
Original Examples Only	52.64%	54.06%	54.42%	54.86%
Rationale Examples Only	51.07%	51.33%	52.01%	54.01%
Faces as Rationales	52.17%	53.08%	53.25%	53.40%
Faces Only	53.26%	54.41%	55.57%	56.14%
Female				
Training Examples per Class	$N = 25$	50	75	100
Ours (Our Annotations)	53.13%	54.51%	55.89%	57.07%
Ours (MTurk Annotations)	53.83%	55.03%	55.85%	56.57%
Original Examples Only	54.02%	55.03%	55.83%	55.99%
Rationale Examples Only	50.06%	50.00%	50.00%	50.00%
Faces as Rationales	53.39%	55.17%	55.77%	56.11%
Faces Only	56.96%	59.05%	60.62%	61.46%

Table 6.2: Hot or Not Results - The performance of our approach applied to the Hot or Not data set, at four different training set sizes, along with the performance of several baselines. For males, our method with our own annotations beats MTurk annotations and each baseline. For females, however, the “Faces Only” baseline beats our performance.

6.2.3 Results of Hot or Not Experiment

Our tests show a significant improvement in classification performance from our approach over each baseline in the case of classifying males based on our own annotations (rather than the Mechanical Turk annotations). However, the improvement is much more limited when the Mechanical Turk annotations are used. In fact, while the MTurk annotations give better classification performance than the “original examples only” and “rationale examples only” baselines, they do not beat the “faces only” baseline.

In the case of females, the improvement in classification performance from rationales over the traditional baseline is much less significant than for the males. Furthermore, the “faces only” baseline has much stronger performance than our approach.

See table 6.2 for detailed results.

6.2.4 Discussion

We will discuss the performance of using our rationale-based approach to classify the appearance of humans and the discrepancies between our attempts at doing so for males and females.

6.2.4.1 Success of our Approach

In table 6.2, it’s obvious that our rationales (though not MTurk rationales) were very helpful in classifying males. In fact, with just $N = 25$ training examples per class, our approach beats the performance of the traditional approach with $N = 100$ training examples per class (55.40% vs. 54.92%). Neither of the face baselines had particularly impressive performance compared to our approach, either. This lends credence to our intuition that rationales can be essential when it comes to training a classifier on such a subjective task as deciding how attractive a person is.

Although rationales do carry a time penalty per annotation, we expect the cost of giving an image a rationale annotation in addition to a class label to be well under four times the time it takes to classify it alone. This makes this result in which rationales beat the baseline with just a quarter of the number of training examples (25 versus 100) a “win” in terms of annotation time as well as number of training examples. For many tasks, for example video tasks where a rationale could be a time segment, the marginal time cost of giving a rationale in addition to a class label could become negligible.

On the other hand, with females, the rationales helped very little: at $N = 100$ training examples per class the performance difference was only about 1% over the traditional, “original examples only” baseline (57.07% vs. 55.99%). Worse still, the “faces only” baseline beat it handily. This leads us to believe that, in females, the face seems much more helpful in deciding attractiveness than it is in males.

6.2.4.2 Differences in Classifying Males and Females

We can speculate on the reasons for the performance differences in each of these different approaches and baselines between the two genders. We notice that rationales weren’t nearly as helpful to classifying females as they were to classifying males. One possible explanation for this phenomenon is that the images of females seem to be tighter shots than those of males, so there are fewer background features to distract the classifier from the human in the image, diminishing the importance of the rationale.

Even more striking is the difference in performance of the face detection baselines for each gender. Classification of females is helped tremendously by face detection (see “Faces Only” baseline in table 6.2), while the effect wasn’t nearly as noticeable for classification of males. One way in which the classification of males might lose out from removing all non-facial information (as the “faces only” baseline does) is in the generally well-toned, shirtless upper bodies of many of the “Hot” males, and the lack of this in the “Not” males. In contrast, the difference in body type of Hot females and Not females comes primarily in the form of weight, rather than different textures in the skin. Because it is difficult to create local features such as SIFT descriptors that convey image-level information like body shape, it’s possible that in our representation, the body is less useful for differentiating Hot females from Not females, hence the greater improvement from only using their faces. An extension of our approach might consider more global models of appearance or shape; however, this would require a different strategy for introducing rationales into the classifier.

6.2.4.3 Low Overall Performance

In all tests of our approach and baselines, the performance never exceeded 62% classification performance, and the most naïve baseline, a coin toss, would give 50% classification performance as this is a binary classification problem. It’s clear that this is a difficult task to learn. There is room for improvement in several areas. With a higher number training examples, performance could easily go up for many of these tests, as performance is clearly trending upwards as the size of the training set increases in most cases (including our approach, for both male and female). A stronger image representation might also improve the performance of both our approach and the baselines. This could include, for example, multi-scale dense SIFT descriptors, other local feature representations like SURF or Gist, spatial pyramid matching, and color bins.

6.2.5 Classification Performance per Image

Noticing large per-image differences in our approach’s performance relative to the baseline, we show the images from this data set for which our approach helped and hurt classification performance the most (see tables 6.3 and 6.4).

6.2 Hot or Not




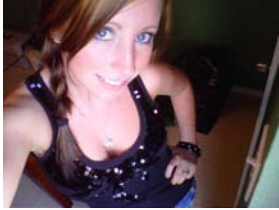
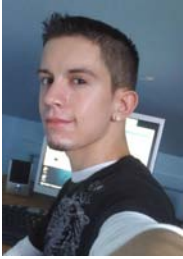
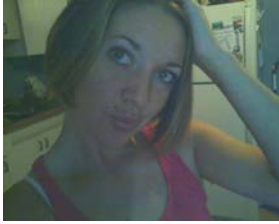
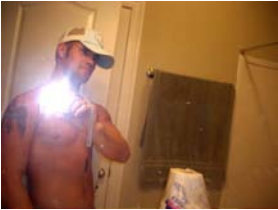

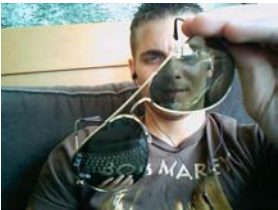

Male	Ours	Base	Female	Ours	Base
	Not 27.60%	12.63%		Hot 61.50%	16.43%
	Not 31.65%	18.26%		Hot 80.60%	22.54%
	Hot 71.60%	42.97%		Hot 52.61%	15.12%
	Hot 62.65%	37.92%		Not 44.70%	14.21%
	Hot 68.25%	43.30%		Hot 42.49%	15.02%

Table 6.3: Best Performance Improvement - Hot or Not - These are the five images of each gender in our data set for which our method gave the greatest classification performance improvement (based on percent gain) over our baseline test without rationales, and the correct classification rate of our approach (column “Ours”) and the baseline (column “Base”). The classification rates are the fraction of times the class of the image was predicted correctly over 2000 random training set splits with training set size $N = 25$.


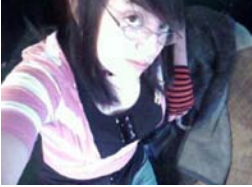








Male		Ours	Base	Female		Ours	Base
	Not	6.31%	32.31%		Not	2.95%	62.20%
	Not	3.85%	18.67%		Hot	4.36%	36.68%
	Not	4.85%	20.53%		Hot	7.80%	44.07%
	Hot	6.25%	21.96%		Not	7.24%	40.31%
	Not	15.25%	47.42%		Not	8.03%	42.93%

Table 6.4: Worst Performance Loss - Hot or Not - These are the five images of each gender in our data set for which our method gave the worst classification performance loss (based on percent loss) relative to our baseline test without rationales, and the correct classification rate of our approach (column “Ours”) and the baseline (column “Base”) over 2000 trials. The classification rates are the fraction of times the class of the image was predicted correctly over 2000 random training set splits with training set size $N = 25$.

7

Conclusions

We have presented a new way to look at supervised learning of image classes: by using not only the “what” of an annotator’s classification, but also the “why”. We have found that asking an annotator to not only label an image by its class but also by drawing a polygon around the region or regions that were most influential in his or her choice can be significantly useful in multiple domains, including creating binary classifiers of scene categories and classifying images of humans as attractive or unattractive. These results seem to suggest that this new approach to image classification could be useful in many other domains as well, especially perceptive or subjective tasks with images that have sparsely distributed interesting features and those that involve a subjective decision by the annotator.

References

- [1] O. ZAIDAN, J. EISNER, AND C. PIATKO. **Using Annotator Rationales to Improve Machine Learning for Text Categorization.** In *Proceedings of the North American Chapter of the Association for Computational Linguistics - Human Language Technologies Conference*, 2007. 3, 6, 9, 11, 12, 34
- [2] Caltech 256 Image Database. <http://www.vision.caltech.edu/ImageDatasets/Caltech256/>, 2007. 4
- [3] The PASCAL Visual Object Classes Challenge 2007. <http://www.pascal-network.org/challenges/VOC/voc2007/workshop/index.html>, 2007. 4
- [4] B. RUSSELL, A. TORRALBA, K. MURPHY, AND W. FREEMAN. **LabelMe: a Database and Web-Based Tool for Image Annotation.** Technical report, MIT, 2005. 4
- [5] J. DENG, W. DONG, R. SOCHER, L.-J. LI, K. LI, AND L. FEI-FEI. **ImageNet: A Large-Scale Hierarchical Image Database.** 2009. 4, 5
- [6] I. LAPTEV, M. MARSZALEK, C. SCHMID, AND B. ROZENFELD. **Learning Realistic Human Actions from Movies.** 2008. 4
- [7] J. YUAN, B. RUSSELL, C. LIU, AND A. TORRALBA. **LabelMe Video: Building a Video Database with Human Annotations.** 2009. 4
- [8] AUDE OLIVA AND ANTONIO TORRALBA. **Scene-Centered Description from Spatial Envelope Properties.** In *2nd Workshop on Biologically Motivated Computer Vision, Lecture Notes in Computer Science*, pages 263–272. Springer-Verlag, 2002. 4, 5
- [9] SVETLANA LAZEBNIK, CORDELIA SCHMID, AND JEAN PONCE. **Beyond Bags of Features: Spatial Pyramid Matching for Recognizing Natural Scene Categories.** In *CVPR*, pages 2169–2178, 2006. 4, 5
- [10] R. YAN, J. YANG, AND A. HAUPTMANN. **Automatically Labeling Video Data Using Multi-Class Active Learning.** 2003. 5
- [11] A. KAPOOR, K. GRAUMAN, R. URTASUN, AND T. DARRELL. **Active Learning with Gaussian Processes for Object Categorization.** 2007. 5

-
- [12] S. VIJAYANARASIMHAN AND K. GRAUMAN. **What's It Going to Cost You?: Predicting Effort vs. Informativeness for Multi-Label Image Annotations.** 2009. 5
 - [13] E. CHANG, S. TONG, K. GOH, AND C. CHANG. **Support Vector Machine Concept-Dependent Active Learning for Image Retrieval.** In *IEEE Transactions on Multimedia*, 2005. 5
 - [14] G. QI, X. HUA, Y. RUI, J. TANG, AND H. ZHANG. **Two-Dimensional Active Learning for Image Classification.** 2008. 5
 - [15] B. COLLINS, J. DENG, K. LI, AND F-F. LI. **Towards Scalable Dataset Construction: An Active Learning Approach.** 2008. 5
 - [16] L. VON AHN AND L. DABBISH. **Labeling Images with a Computer Game.** 2004. 5
 - [17] L. VON AHN, R. LIU, AND M. BLUM. **Peekaboom: A Game for Locating Objects in Images.** 2006. 5
 - [18] A. SOROKIN AND D. FORSYTH. **Utility Data Annotation with Amazon Mechanical Turk.** In *Proceedings of the CVPR Workshop on Internet Vision*, 2008. 5
 - [19] E. CHANG, S. TONG, K. GOH, AND C.-W. CHANG. **Support Vector Machine Concept-Dependent Active Learning For Image Retrieval.** *IEEE Transactions on Multimedia*, 2005. 5
 - [20] H. RAGHAVAN, O. MADANI, AND R. JONES. **InterActive Feature Selection.** 2005. 5
 - [21] G. DRUCK, B. SETTLES, AND A. MCCALLUM. **Active Learning by Labeling Features.** In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing*, 2009. 5
 - [22] D. LOWE. **Distinctive Image Features from Scale-Invariant Keypoints.** 60(2), 2004. 8
 - [23] A. VEDALDI AND B. FULKERSON. **VLFeat: An Open and Portable Library of Computer Vision Algorithms.** <http://www.vlfeat.org/>, 2008. 8
 - [24] L. FEI-FEI AND P. PERONA. **A Bayesian Hierarchical Model for Learning Natural Scene Categories.** In *CVPR '05: Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, 2, pages 524–531, June 2005. 16, 17, 20
 - [25] G. DORKO AND C. SCHMID. **Selection of Scale-Invariant Parts for Object Class Recognition.** In *ICCV*, pages 634–650, 2003. 29, 31