Classification of Trash for Recyclability Status

Mindy Yang
Stanford University
mindyang@stanford.edu

Gary Thung
Stanford University
gthung@stanford.edu

***Abstract*—A computer vision approach to classifying garbage into recycling categories could be an efficient way to process waste. The objective of this project is to take images of a single piece of recycling or garbage and classify it into six classes consisting of glass, paper, metal, plastic, cardboard, and trash. We also create a dataset that contains around 400-500 images for each class, which was hand collected. We plan to release this dataset for the public. The models used are support vector machines (SVM) with scale-invariant feature transform (SIFT) features and a convolutional neural network (CNN). Our experiments showed that the SVM performed better than the CNN; however, the CNN was not trained to its full capability due to trouble finding optimal hyperparameters.**

I. Introduction

Recycling is necessary for a sustainable society. The current recycling process requires recycling facilities to sort garbage by hand and use a series of large filters to separate out more defined objects. Consumers also can be confused about how to determine the correct way to dispose of a large variety of materials used in packaging. Our motivation was to find an automatic method for sorting trash. This has the potential to make processing plants more efficient and help reduce waste, as it is not always the case that the employees sort everything with 100% accuracy. This will not only have positive environmental effects but also beneficial economic effects.

In order to mimic a stream of materials at a recycling plant or a consumer taking an image of a material to identify it, our classification problem involves receiving images of a single object and classifying it into a recycling material type. The input to our pipeline are images in which a single object is present on a clean white background. We then use an SVM and CNN to classify the image into six categories of garbage classes. By using computer vision, we can predict the category of garbage that an object belongs to based on just an image.

II. Related Work

Previously, there have been many support vector machine and neural network based image classification research projects. However, there are none that pertain specifically to trash classification.

In the realm of image classification, one well-known and highly capable CNN architecture is AlexNet [1], which won the 2012 ImageNet Large-Scale Visual Recognition Challenge (ILSVRC). The architecture is relatively simple and not extremely deep, and is, of course, known to perform well. AlexNet was influential because it started a trend of CNN approaches being very popular in the ImageNet challenge and becoming the state of the art in image classification.

The most similar project we have found was a project from the 2016 TechCrunch Disrupt Hackathon [2] in which the team created "Auto-Trash", an auto-sorting trashcan that can distinguish between compost and recycling using a Raspberry Pi powered module and camera. Their project was built using Google's TensorFlow and it also includes hardware components. Something to note about Auto-Trash is that it only classifies whether something is compost or recycling, which is a simpler than having five or six classes.

Another trash related project was a smartphone application designed to coarsely segment a pile of garbage in an image [3]. The goal of the application is to allow citizens to track and report garbage in their neighborhoods. The dataset used was obtained through Bing Image Search and the authors extracted patches from the images to train their network. The authors utilized a pre-trained AlexNet

model and obtained a mean accuracy of 87.69%. The authors did well to take advantage of a pre-trained model to improve generalization.

Other recycling based classification problems used physical features of an object. In 1999, a project from Lulea University of Technology [4] worked on recycling metal scraps using a mechanical shape identifier. They used chemical and mechanical methods such as probing to identify the chemical contents and current separation. This paper's mechanical approach provides interesting advancement strategies for our project.

Another more image based classification of materials was performed on the Flickr Materials Database [5]. The team used features such as SIFT, color, microtexture and outline shape in a Bayesian computational framework. This project is similar to ours in that it attempts to classify images based on material classes. However, the dataset used is different than ours in that the images are untarnished materials with no logos or deformation.

III. Dataset and Data Collection

The data acquisition process was done by hand by us because there are no publicly available datasets pertaining to garbage materials. Originally we were using the Flickr Material Database and images from Google Images. However, these images do not accurately represent the state of recycled goods after more research on recycling plants and the state of recycled goods. For example, the images in the Flickr Material Database present materials in a pristine and undamaged state. This is unlikely in recycled materials treated as waste because they are dirty, ruffled, crumpled, etc.

Therefore, we hand collected our own dataset of images, which we plan on making a public dataset. The dataset contains images of recycled objects across six classes with about 400-500 images each (besides the "trash" class which only has about 100 images), totaling about 2,400 images. The data acquisition process involved using a white posterboard as a background and taking pictures of trash and recycling around Stanford, our homes, and our relatives' homes. The lighting and pose for each photo is not the same, which introduces variation in the dataset. The figures below show example images from the six classes. Data augmentation techniques

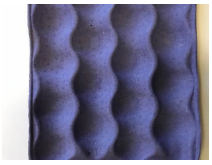

Fig. 1. Paper

Fig. 2. Glass

Fig. 3. Plastic

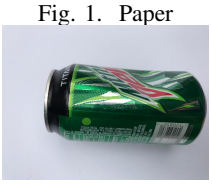

Fig. 4. Metal

Fig. 5. Cardboard

Fig. 6. Trash

were performed on each image because of the small size of each class. These techniques included random rotation of the image, random brightness control of the image, random translation of the image, random scaling of the image, and random shearing of the image. These image transformations were chosen to account for the different orientations of recycled material and to maximize the dataset size. We also performed mean subtraction and normalization.

IV. Model and Methods

A. Support Vector Machine

An SVM was used for the first run through for the classification of trash into recycling categories. The SVM was chosen because it is considered one of the best initial classification algorithms and is not as complicated compared to a CNN.

The SVM classifies items by defining a separating hyperplane for multidimensional data. The hyperplane that the algorithm attempts to find is the hyperplane that gives the largest minimum distance to the training examples. More specifically, an SVM's optimization objective is

$$\min_{\gamma,w,b} \frac{1}{2}||w||^2$$

$$\text{s.t } y^{(i)}(w^T x^{(i)} + b) \geq 1, i = 1, ..., m$$

where w, b are parameters of our hypothesis function, $y^{(i)}$ represents the label for a specific example, $x^{(i)}$ is the i^{th} example out of m, and γ is the minimum geometric margin of all training examples. For a multiclass SVM, a common method is a one versus all classification where the class is chosen

based on which class model classifies the test datum with greatest margin.

The features used for the SVM were SIFT features. On a high level, the SIFT algorithm finds blob like features in an image and describes each in 128 numbers.

Specifically, the SIFT algorithm passes a difference of Gaussian filter that varies σ values as an approximate for Laplacian of Gaussian. The σ values act to detect larger and smaller areas of an image. Then images are then searched for local extrema over scale and space. A pixel in an image is compared with neighbors of varying scale. If the pixel is a local extrema, it is a potential key point. This also means that the keypoint is best represented in that specific scale. Once potential key points are found they have to be refined through Taylor series expansion and thresholding. Then orientation is assigned to each keypoint to achieve invariance to image rotation. The keypoint is rotated in 360 directions plotted like a histogram in 36 bins (10 degrees per bin) based on the gradient magnitude at certain rotations. The keypoint is chosen to be the rotation with the highest number of values in a bin. After the keypoint is found, a 16x16 neighborhood around the keypoint is taken. It is then divided into 16 sub-blocks of 4x4 size. For each sub-block, 8 bin orientation histogram is created. So a total of 128 bin values are available. SIFT features are powerful

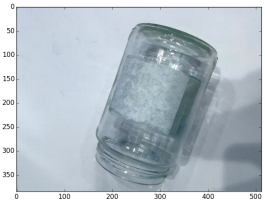

Fig. 7. Original image

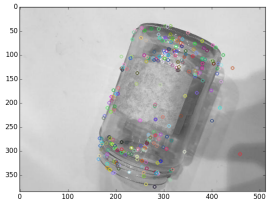

Fig. 8. SIFT keypoints detected

because they are invariant to scale, noise and illumination which is perfect for recycling classification. Most recycling objects are not extremely different looking but are variations in size and color.

Then bag of features was applied. The SIFT descriptors for the training images were clustered by the k-means algorithm where k was the number of training examples. Then for each new test example, the SIFT features are pulled and a histogram of values based on the original clustering is used as the data point for the dataset. This greatly reduces the required SVM training time since an image is reduced to a histogram.

B. Convolutional Neural Network

We use the Torch7 framework for Lua to construct our CNN. We implemented an eleven layer CNN that is very similar to AlexNet. Our network is smaller than AlexNet (using $\frac{3}{4}$ of the amount of filters for some convolutional layers) because of computational constraints.

- Layer 0: Input image of size 256x256
- Layer 1: Convolution with 96 filters, size 11x11, stride 4, padding 2
- Layer 2: Max-Pooling with a size 3x3 filter, stride 2
- Layer 3: Convolution with 192 filters, size 5x5, stride 1, padding 2
- Layer 4: Max-Pooling with a size 3x3 filter, stride 2
- Layer 5: Convolution with 288 filters, size 3x3, stride 1, padding 1
- Layer 6: Convolution with 288 filters, size 3x3, stride 1, padding 1
- Layer 7: Convolution with 192 filters, size 3x3, stride 1, padding 1
- Layer 8: Max-Pooling with a size 3x3, stride 2
- Layer 9: Fully Connected with 4096 neurons
- Layer 10: Fully Connected with 4096 neurons
- Layer 11: Fully Connected with 5 neurons
- Result: Non-normalized log softmax scores, 5 classes

V. Experiments

A. Support Vector Machines

For the SVM, a radial basis kernel was chosen. The kernel is defined as

$$K(x, x') = exp(-\frac{||x - x'||^2}{2\sigma^2})$$

Radial basis kernels are often the best for image datasets. We experimented with other kernels, such as the linear kernel and polynomial kernel, but those did not perform as well.

The C parameter of the SVM was set to 1000. This parameter tells the SVM optimization how much to avoid misclassifying each training example.

A low C parameter did not work on this dataset because the SVM simply returned the same label for all of the data. This value was found from an exploration of a range of numbers.

Gamma was set to an intermediate value of 0.5 as to not require too extreme margins or too small margins.

B. Convolutional Neural Network

The CNN was trained with a train/val/test split of 70/13/17, an image size of 256x256, 60 epochs, a batch size of 32, a learning rate of 5e-8, 5e-1 weight decay every 5 epochs, an L2 regularization strength of 7.5e-2, and Kaiming weight initialization [6]. We did not use the same hyperparameters that AlexNet used because of the differing tasks at hand (ImageNet contains about 1.3 million images). Many hyperparameters were experimented with and these were the ultimate hyperparameters we ended up with.

We encountered trouble training the neural network, as it would not learn. We chose to omit the "trash" class images because there were only about $\frac{1}{5}$ of the images compared to the other classes because they would create an imbalance in the dataset.

VI. Results

A. Support Vector Machines

The SVM achieved better results than the CNN. It achieved a test accuracy of 63% using a 70/30 training/testing data split. The training error was 30%. The SVM is a relatively simpler algorithm than the CNN, which may attribute to its success in this task.

material	precision	recall
glass	0.55	0.60
paper	0.80	0.70
cardboard	0.62	0.66
plastic	0.61	0.69
metal	0.70	0.59
trash	0.24	0.35

B. Convolutional Neural Network

As stated in the experiments section, we had trouble with training the network. The network seemed to not learn, as the test accuracy we achieved in the experiment described was only 22%. This is barely better than random classification and it tells us that the hyperparameters are not working well, or the model is too complex or too simple.

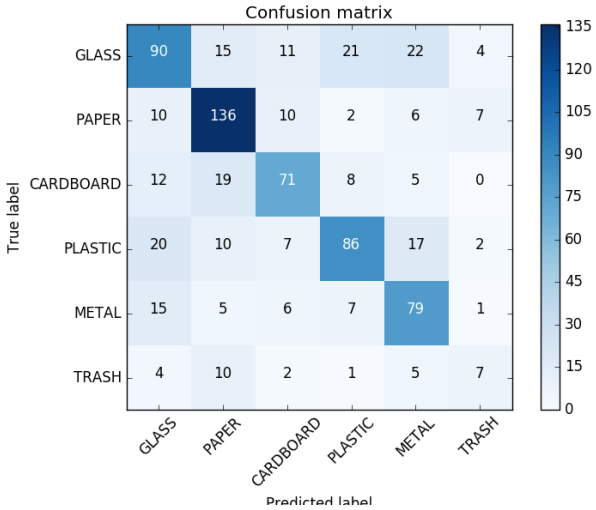

Fig. 9. Confusion matrix from the SVM run on a smaller version of the dataset.

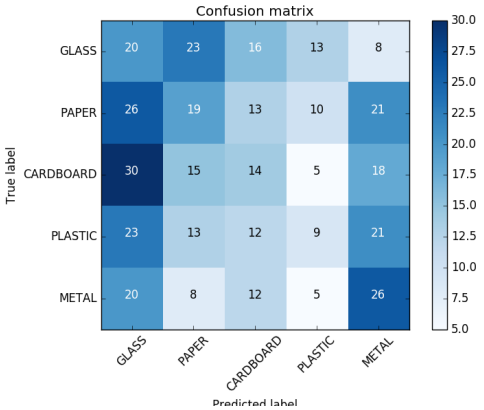

Fig. 10. Confusion matrix from the CNN run on the test split.

material	precision	recall
glass	0.25	0.17
paper	0.21	0.29
cardboard	0.17	0.21
plastic	0.12	0.21
metal	0.37	0.28

We saw the same problem of the network not learning on earlier attempts at training the network with a variety of hyperparameters. Previously, we used an image size of 384x384, batch size of 50, and no weight initialization beyond random. Thus,

we reduced the image size to reduce complexity, reduced the batch size to be more appropriate for the dataset size, and used a weight initialization technique to improve learning. We believe that the CNN's inability to learn is related to the hyperparameters being suboptimal, as the loss is erratic and would indicate that the learning rate may be too aggressive, which would cause it to fluctuate up and down, and not decrease at a consistent rate. The same applies for the training and validation accuracy not increasing and also exhibiting erratic behavior.

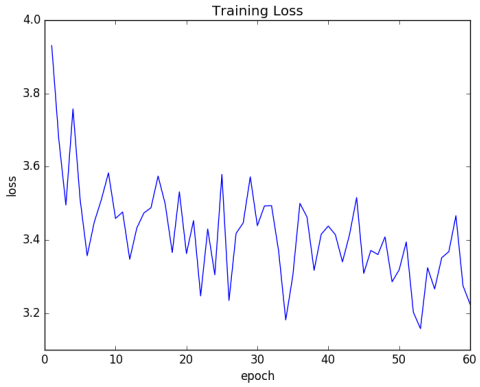

Fig. 11. Training loss for the CNN.

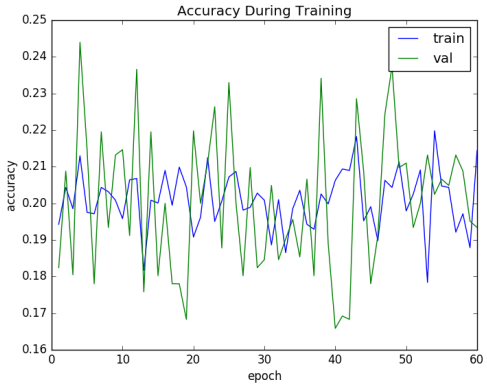

Fig. 12. Training and validation accuracy for the CNN.

VII. Discussion

The SVM performed better than the CNN, which is not what we expected. Given that the SVM is a simpler algorithm, that is likely the reason for its superior performance out of the box. Neural networks require a substantial amount of time to train and tune to achieve optimal performance. Based on previous research results with neural networks, they have a higher ceiling for potential.

We gained more respect for all of the publicly available datasets. The dataset collection was extremely tedious and at times dirty. We will be gathering some more images and releasing our dataset for public use.

To improve our CNN results, we could have collected much more data if the time frame was longer. We attempted to maximize the data we had through augmentation. Along with that, more thorough hyperparameter search could be performed.

VIII. Conclusion

The classification of trash into various recycling categories is possible through machine learning and computer vision algorithms. One of the biggest pain point is the wide varieties of possible data (i.e. any object can be classified into one of the waste or recycling categories). Therefore, in order to create a more accurate system, there needs to be a large and continuously growing data source.

IX. Future Work

First and foremost, we want to continue working on the CNN to figure out why it did not train well and to train it to achieve a good accuracy. We expect that it should perform significantly better than the SVM classifier. Furthermore, we would like to extend this project to identify and classify multiple objects from a single image or video. This could help recycling facilities more by processing a stream of recycling rather than single objects.

Another important addition could be multiple object detection and classification. This would improve large scale classification of recycling materials.

Finally, we want to continue expanding our dataset by adding more photos, especially in the trash class, and possibly more classes, and then finally releasing it.

References

[1] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in Neural Information Processing Systems 25*, F. Pereira, C. J. C. Burges, L. Bottou, and K. Q. Weinberger, Eds. Curran Associates, Inc., 2012, pp. 1097–1105. [Online]. Available: http://papers.nips.cc/paper/4824-imagenet-classification-with-deep-convolutional-neural-networks.pdf

[2] J. Donovan, "Auto-trash sorts garbage automatically at the techcrunch disrupt hackathon." [Online]. Available: https://techcrunch.com/2016/09/13/auto-trash-sorts-garbage-automatically-at-the-techcrunch-disrupt-hackathon/

[3] G. Mittal, K. B. Yagnik, M. Garg, and N. C. Krishnan, "Spotgarbage: Smartphone app to detect garbage using deep learning," in *Proceedings of the 2016 ACM International Joint Conference on Pervasive and Ubiquitous Computing*, ser. UbiComp '16. New York, NY, USA: ACM, 2016, pp. 940–945. [Online]. Available: http://doi.acm.org/10.1145/2971648.2971731

[4] S. Zhang and E. Forssberg, "Intelligent liberation and classification of electronic scrap," *Powder technology*, vol. 105, no. 1, pp. 295–301, 1999.

[5] C. Liu, L. Sharan, E. H. Adelson, and R. Rosenholtz, "Exploring features in a bayesian framework for material recognition," in *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*. IEEE, 2010, pp. 239–246.

[6] K. He, X. Zhang, S. Ren, and J. Sun, "Delving deep into rectifiers: Surpassing human-level performance on imagenet classification," in *The IEEE International Conference on Computer Vision (ICCV)*, December 2015.