Active Learning for Image Recognition using a Visualization-Based User Interface

Christian Limberg[1,2], Kathrin Krieger[1], Heiko Wersing[2], and Helge Ritter[1]

[1] Bielefeld University, Neuroinformatics Group, Universitätsstraße 25, 33615 Bielefeld - Germany {climberg,kkrieger,helge}@techfak.uni-bielefeld.de
[2] HONDA Research Institute Europe GmbH, Carl-Legien-Straße 30, 63073 Offenbach - Germany heiko.wersing@honda-ri.de

Abstract. This paper introduces a novel approach for querying samples to be labeled in active learning for image recognition. The user is able to efficiently label images with a visualization for training a classifier. This visualization is achieved by using dimension reduction techniques to create a 2D feature embedding from high-dimensional features. This is made possible by a querying strategy specifically designed for the visualization, seeking optimized bounding-box views for subsequent labeling. The approach is implemented in a web-based prototype. It is compared in-depth to other active learning querying strategies within a user study we conducted with 31 participants on a challenging data set. While using our approach, the participants could train a more accurate classifier than with the other approaches. Additionally, we demonstrate that due to the visualization, the number of labeled samples increases and also the label quality improves.

Keywords: Active Learning, Classification, Pattern Recognition, Image Recognition, Object Recognition, User Interface, Visualization, Dimension Reduction

1 Motivation

In a classification task, there are machine learning models that can be trained incrementally and samples can be labeled step-wise by the user. Active learning [14] is an efficient training technique, where the samples, which are predicted to deliver the highest improvement for the classifier, are chosen for being labeled. There are several approaches for selecting the samples to be queried. However, it depends on the actual data which approach yields the best accuracy [16].

Having this in mind, we try to find a more efficient way for applying active learning. The common practice is to ask the human for a label of one single sample at a time [15]. Since this is a monotonous task and therefore often leads to mislabeled samples, we want to intervene already at this point by using a labeling user interface which is not only capable of boosting the performance of the classifier and increase the number of labeled samples, but also gives the human a more pleasurable experience while training the classifier. Another goal

is to give the human a better idea about the inner representation of the trained model. This insight may lead to a better understanding where strengths and weaknesses of a feature representation are. To facilitate human labeling of high-dimensional samples, we use dimension reduction techniques to visualize the data in a 2D feature embedding space. We use this for improving active querying in an image recognition task.

There are some approaches towards machine learning using such a visualization. Recently, Cavallo et al. [1] introduced an approach for not only visualizing high-dimensional data, but also changing both the data in the feature embedding space and in high-dimensional space. For instance, after changing data in feature embedding space it can be explored what effect this has in the high-dimensional data and vice versa. Iwata et al. [6] introduced an approach where the user can relocate the data in a visualization to be more representative for him. This can be useful if data is clustered in different categories and a category should be located in one region of the visualization space. It is also useful for ordering data, if it has a natural ordering like numbers or letters.

More related to active learning, there are approaches using scatter plots for visualizing data to facilitate labeling. Huang et al. [5] improved the labeling process of text documents showing the human visualizations of the feature space. The text data is visualized by t-SNE [13], force-directed graph layout and chord diagrams. Hongsen et al. [9] used semi-supervised metric learning to train a visualization of video data. In both approaches, the data is displayed next to the scatter plot for labeling. The querying of samples is done manually by the user, so there is no active learning strategy involved directly, which we want to accomplish for image recognition.

We introduce an active querying technique which utilizes the visualization and enables an efficient training by finding bounding-box views in the visualization for labeling images. Within a user study on a challenging outdoor object data set, we show that using a visualization is favorable and that using our adaptive interface together with the proposed querying method is more efficient than state-of-the-art approaches.

2 Active Learning

Active learning is an efficient technique for training a classifier incrementally. One variant of it is pool-based active learning, where the features $\mathbf{X}$ with labels $\mathbf{Y}$ are divided in an unlabeled pool $\mathbf{U}$ and a labeled pool $\mathbf{L}$. A querying function selects the most relevant samples from $\mathbf{U}$ to be labeled by an oracle, which is in most cases a human annotator. As the training progresses, samples from the unlabeled pool $\mathbf{U}$ are labeled and put in the labeled pool $\mathbf{L}$. Simultaneously, the classifier c is trained online with the new labeled samples.

There were many research contributions in the past proposing querying methods for high performance gain of the classifier. An often used approach is Uncertainty Sampling (US), originally proposed by Lewis et al. [8]. In US the classifier's confidence estimation c_p of the samples from the unlabeled pool are used

to select those with the lowest certainty for querying: $argmin_{u \in \mathbf{U}} c_p(u)$. Another technique is query by committee (QBC) [17], where the query is chosen that maximizes the disagreement of the classifiers. In our evaluation we use the vote entropy for measuring the disagreement of classifiers: $argmax_{u \in \mathbf{U}} - \sum_i \frac{V(y_i)}{C} log \frac{V(y_i)}{C}$ where y_i is a particular label and $V(y_i)$ is the number of classifiers voted for this label, C is the number of classifiers in the committee. In our evaluation we chose a linear Support Vector Machine, a Decision Tree and Logistic Regression as a committee of diverse classifiers.

3 Dimension reduction for visualization

There are many dimension reduction approaches to visualize a high-dimensional feature space in lower dimensions. Their training is usually unsupervised. An early approach is Principal Component Analysis (PCA) [4], where a small set of linearly uncorrelated variables having the highest variance in the data, called principal components, are extracted. Multidimensional Scaling (MDS) [19] is a technique for dimension reduction, which preserves the spatial relation of the high-dimensional data in the lower-dimensional space. A Self Organizing Map (SOM) [7], introduced by Kohonen in 1982, can be used for dimension reduction. By applying competitive learning SOMs can preserve topological properties in the lower dimensional map.

In 2008, van der Maaten et al. proposed t-SNE [13], which is a variant of Stochastic Neighbor Embedding (SNE) [3]. By modeling data points as pairwise probabilities in both the original space and the embedding, using a gradient decent method to minimize the sum of Kullback-Leibler divergences, it is possible to create an embedding of high quality. Especially if there are classes with different variances in high-dimensional space, t-SNE delivers reasonable results. Our preliminary experiments also show, that t-SNE is delivering best results compared to PCA and MDS for image data where classes consist of objects showed from different viewing positions, like in the OUTDOOR data set [12] that we will also use in our evaluation. Because of these advantages, we use t-SNE as a dimension reduction technique in our experiments, but basically every other approach can be used as well.

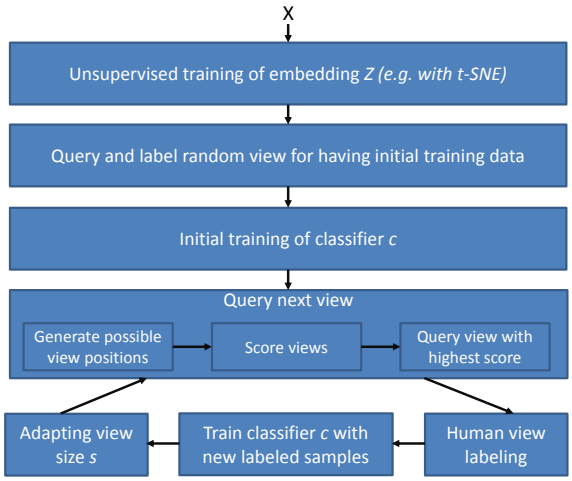

Fig. 1: General workflow diagram describing active learning using a visualization.

4 Adaptive Visualization View Querying (A2VQ)

The underlying idea is to query the samples within a bounding-box view of the visualization which we denote as a view $\mathbf{v}$. The goal of our approach is to query the optimal view for labeling of its enclosed samples.

In the following we introduce the Adaptive Visualization View Querying (A2VQ) approach for querying in active learning using an adaptive visualization. The overall workflow is illustrated in Fig. 1. First, we use the t-SNE algorithm to reduce the high dimensionality of $\mathbf{X}$ (usually a high dimensional feature description of an image using e.g. a CNN) to 2D for visualization. We normalize the output by applying feature scaling so that values of each of the two dimensions are between 0 and 1, naming this normalized embedding feature space $\mathbf{Z}$. In the following we refer $\mathbf{Z}_i$ as the visualization of sample $\mathbf{U}_i$.

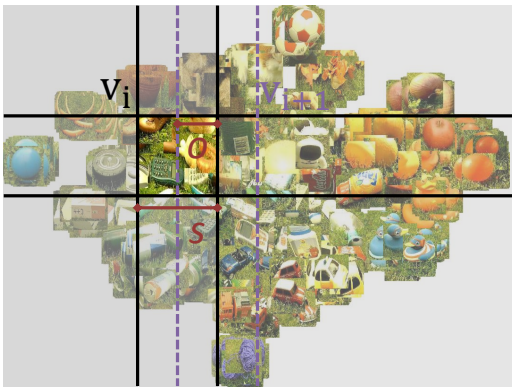

Fig. 2: t-SNE visualization of 50 objects from the OUTDOOR data set with illustrated sliding window approach. In one iteration of sliding window, all views of the visualization are scored by A2VQ's scoring function. The possible views are generated by moving the squared template with side length s in overlap o steps from the upper left to the bottom right corner. The view with highest score is queried for labeling and displayed in our web-based user interface.

Since we assume to have no label information at the beginning, active training starts with an empty $\mathbf{L}$. So labeling of one or more randomly generated views is necessary to initially train a classifier for our approach. Then confidences for samples of $\mathbf{U}$ are calculated by the classifier, used to query the optimal view (described in detail in the next section). The queried view can be labeled e.g. by a user with our proposed user interface. Then the classifier is trained incrementally with the newly labeled samples. After this training epoch, a new optimal view is queried with the retrained classifier and the process repeats.

We think, a querying method is necessary for an efficient labeling because a visualization of more complex data sets can be confusing for the human as there are too many classes and the images are highly overlapping as one can see in Fig. 2. Also we want to be able to actively query the samples which the classifier demands for efficient training.

4.1 Visualization View Querying

To query the optimal view we use a sliding window technique to cycle through a grid of possible bounding-box views that arises from a view size s and overlap

amount o. The first view is positioned at $(0, 0)$ in $\mathbf{Z}$. By shifting the square $s - o$ in each dimension (illustrated in Fig. 2), there is a total number of $(1 + \frac{1-s}{s-o})^2$ views to be evaluated. We therefore calculate a scoring function $r(\mathbf{v})$ for each view:

$$r(\mathbf{v}) = \frac{\sum_{u \in \mathbf{U_v}} (1 - c_p(u))}{m} \tag{1}$$

where $\mathbf{U_v}$ are the samples lying in the particular view, $c_p(u)$ is the classifier's confidences of the most certain class for sample u and m is the number of samples in the view with the most enclosed samples. By dividing by m not only the classifier's confidences of the view's samples are taken into account, but also the number of samples in the view. We do this for not querying views with few outlier samples with low confidences, as they can occur for instance at border areas in a t-SNE visualization (see Fig. 2). After calculating r for each view generated by the sliding window approach, the view with the highest score r is queried for labeling.

4.2 User interface

The samples of the optimal view can be labeled with our user interface, also available at github[3] together with all implemented querying techniques. By applying an affine transformation the view is shown in full size with the corresponding sample images as scatter plot symbols. The resulting display is shown in Fig. 3. Due to the visualization most neighboring samples will receive the same label. Interactive selection techniques (see Fig. 3) allow economic labeling of the samples within the view.

Fig. 3: Querying user interface showing a view queried by A2VQ. The user can label samples via selecting their thumbnails by dragging rectangles in the visualization. The class name can be entered in an input formula. With the button *Label* the selected samples are labeled and removed from the view. The button *Remove Selection* removes the rectangles. There are certain possible labeling strategies, like label everything, label only the biggest clusters or label only outliers. With a click on the button *Query next view* the classifier is retrained with the new labeled samples and a new view is queried with A2VQ.

4.3 Adaptive view size

In addition to querying the best view for labeling, there is the question of finding the best view size s. A small s

[3] https://github.com/limchr/A2VQ

would not be efficient for labeling and a too large s makes it impossible for the human to recognize the images because there are too many. We investigated two heuristics for finding a suitable view size.

Number of Classes: In this heuristic we assume that showing the user about $c = 3$ different classes within a view results in best usability. We incrementally increase or shrink s we use a heuristic that is evaluated after each labeled view:

$$s = s + \sigma(\lambda * (c - n)) - 0.5 \tag{2}$$

where λ is the learning rate, n are the number of individual classes in the view after removing outlier classes with less than 5 samples and σ is the sigmoid function. Using the learning rate inside the sigmoid function, which is centered vertically by subtracting 0.5, enables us to incrementally change the view size to match c.

Preliminary (automated) experiments showed, that adjusting view size with upper heuristic converges to a proper view size with $\lambda = 0.05$. However, in our automated experiments we assumed that the user has perfect ability in labeling the samples and that he labels all samples within a view, whereas in our user study we also train ambiguous objects. So we want to give the human the change to skip samples. Since we can not evaluate n then, we used another heuristic for choosing the view size:

Number of Samples: We assume that a view should not have more than $b = 100$ samples so that the user is able to recognize them while using our label interface. To determine the s that fits this assumption, we count the number of samples within all possible views. We sort this array in descending order and choose the highest 20% for calculating a mean, naming it m. We do this for several view sizes $\{0.10, 0.15, 0.20, 0.25, 0.30, 0.35, 0.40\}$ and choose the view size with the minimum $|b - m|$. In our user study we evaluated $s = 0.25$ and chose $o = 0.5s$. A smaller overlap would be possible but requires longer calculation time because more views have to be evaluated while querying.

5 Evaluation

5.1 Experiment

We did a user study for comparing A2VQ to the baselines US, QBC and random querying (RAND).

Participants 31 participants (*gender*: 16 males, 13 females, 2 others. *status*: 27 students, 2 employees, 2 others) joined the study. The median of their age was 28 years. The participants were paid 5€ for completing the whole study which took 30 to 45 minutes. Three of the participants refused the money. The protocol was approved by the Bielefeld University Ethics Committee.

User interfaces In the study, participants labeled images with two different user interfaces. For A2VQ they used the already described user interface (see Fig. 3). Participants were told, that it is not necessary to label all images within one view because we wanted to give them the ability to skip samples in all approaches. If none of the images within a view could be labeled, the view with the next higher score was displayed. A classic user interface was used for labeling with US, QBC and RAND (see Fig. 4). To label an image, participants had to choose a label from the upper left drop down menu and click the *Label* button. If they were not sure about the label of an image, they could click the *Skip* button. After skipping an image we use DBQE [11] to prevent the querying of similar ambiguous images again, to speed up training.

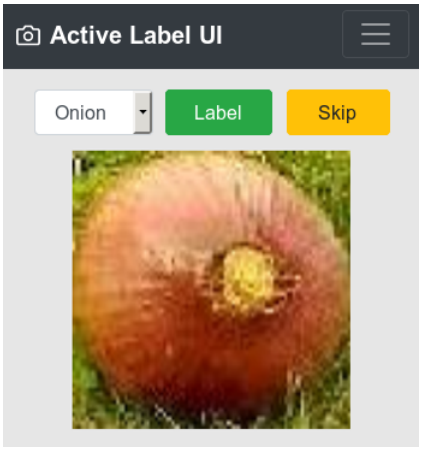

Fig. 4: Classic labeling interface for comparing with the baseline approaches US, QBC and RAND.

Data set We chose the OUTDOOR data set [12] for labeling in the experiment. The data set consists of 5000 images showing objects of 50 classes in a garden environment. Since this are too many classes to be labeled properly within a feasible time, we decided to reduce the data set to only seven classes. To make the labeling challenging for the participants, we selected object classes which might look very similar: *Onion*, *Orange*, *Potato*, *RedApple*, *RedBall*, *Tomato* and *YellowApple*. As a preprocessing step, the objects are cropped using a color segmentation. For feature extraction we used the penultimate layer of the VGG19 CNN [18] trained for the imagenet competition, resulting in a 4096 dimensional feature vector. For evaluation we used a 80/20 train-test split. The test images are used to evaluate the classifier's performance. The images of the train set were presented in the user interfaces and labeled by the participants. We have chosen a 1 nearest Neighbor classifier with the same parameters for all the approaches. For estimating classifier confidences c_p we chose relative similarity [10]. The classifier is trained in an online fashion after each labeled image in the classic labeling interface, or after each labeled image batch in A2VQ.

Task and procedure At the beginning participants signed an informed consent. They read the global task instructions telling them that the main task is to label images to train a service robot to distinguish objects. Afterwards, they performed four experimental trials. They all followed the same procedure. First, participants had read specific task instructions. It contained information about which of the two user interfaces they will use in the following trial and how to interact with it. Before using the user interface for the first time, they watched a short video about the user interface's usage. Thereafter, the trial started and

participants had to label images for five minutes. They were told to be as fast as possible but also as accurate as possible. After five minutes the trial was stopped automatically by the system.

Data recording Whenever a participant labeled an image with any of the tested approaches, several information were saved. We saved the time in milliseconds since the start of the experiment, the index of the labeled image, the given label, the ground truth label and the classifier's 0/1 accuracies on both train and test set.

Experimental design Since each participant labeled once with each approach, they performed four trials in which they labeled the same images. Therefore, it is likely that participants become familiar with the images and improve their labeling performance during the experiment. To avoid such effects having an impact on the analysis, we varied the order of the experimental trials between the 31 participants. There are 24 different possibilities to order four experimental trials. Seven of them were chosen randomly to take place twice resulting in 31 orders which were matched to the participants randomly.

5.2 Analysis

We investigated the impact of the querying approaches A2VQ, US, QBC and RAND on three different parameters. The first one is the *classifier's accuracy* for the test data set. The accuracy's temporal progress and the final accuracy after 5 minutes of training was explored. The second parameter was the *human label quality* which describes how much of the data was labeled correctly by the participants. Finally, we analyzed whether the querying approaches have an impact on the *number of samples* which are labeled during five minutes.

We aimed at analyzing whether there are significant differences in the three parameters influenced by the querying approaches. Therefore, we first checked whether the data meets the assumptions to perform an ANOVA with repeated measures. Inspection of box-plots showed outliers in all three parameters' data. Furthermore the data were not normally distributed as assessed by Shapiro-Wilk's test ($p < .05$). According to this, we performed a two-sided Friedman's test (with $\alpha = .05$) instead of the ANOVA. For each of the three parameters, which showed significant results in Friedman's test, we checked which of the querying approaches differs significantly from each other. Hence, we conducted multiple comparisons with a Bonferroni correction. Statistical tests were conducted with IBM, SPSS Statistics, Version 23.

5.3 Results and discussion

Classifier's accuracy Figure 5 shows the temporal progress of the classifier's accuracy on the test data during training. A2VQ had a slower increase of accuracy in early training while having a higher accuracy at the end (4.8% better than US). The slow rise might be because labeling with A2VQ is comparable

Table 1: Overview of means, medians and standard deviation as well as results of Friedman's test.

	M	Mdn	SD	$\chi^2(3)$	p
classifier's accuracy in %	0.73	0.74	0.15	10.869	.012*
human label quality in %	0.79	0.81	0.11	9.311	.025*
number of labeled samples	148.63	62	171.13	60.650	<.001*

Note: An asterisk marks significant differences between the querying approaches on a level of $\alpha = .05$.

with a depth-first search in a tree. Contrariwise the other approaches are rather comparable with a breadth-first search, having a representation of each object class early in training. Most of the time QBC performed better then US, which performed better than RAND. All baseline approaches started to converge near the end of the experiment.

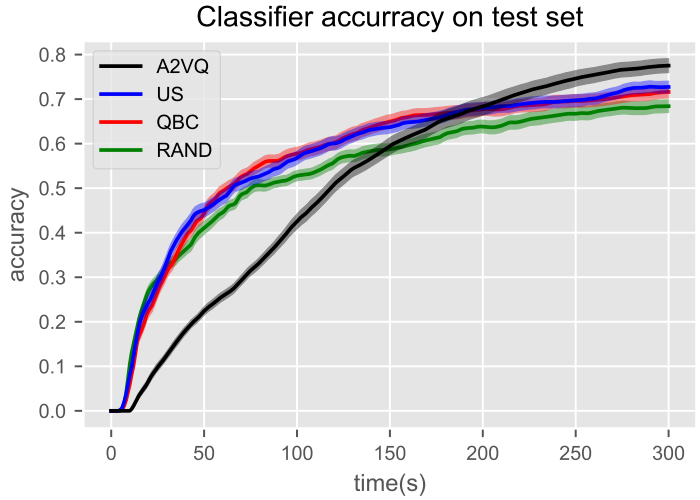

Fig. 5: Classifier's accuracy on held out test set while active training.

Friedman's test, comparing the accuracies of the different approaches after five minutes training, showed significant results. Post hoc tests revealed significant differences between A2VQ and QBC with p=.021 and between A2VQ and RAND with p=.002. This implies A2VQ delivers a better accuracy than RAND and QBC after five minutes training. Even if we did not find any significant differences between A2VQ and US, we can state that in our study A2VQ had the best mean accuracy compared with the other approaches after training the classifier for five minutes.

Human label quality In Fig. 6, a confusion matrix is displayed showing the true labels and the labels given by the participants averaged over all approaches. The

labeling task was challenging for the participants who were not perfect oracles while labeling. This is especially noticeable at classes *RedApple*, *RedBall* and *Tomato* with a label quality of 80% and below.

To compare the label accuracy of the participants between the tested approaches, we performed Friedman's test. Results were significant and, therefore, we performed multiple comparisons with a Bonferroni correction. There were significant differences between A2VQ and all baseline approaches (A2VQ and US with $p = .005$, A2VQ and QBC with $p < .021$, A2VQ and RAND with $p = .030$). Figure 7 demonstrates the results. Using A2VQ results in the best label quality, which is around 4% better than the second best. The reason for this may be an improved human capability to see the objects in context with similar other objects and then to decide. Furthermore the RAND querying approach results in the second best label quality. This may lead to the assumption that classifier's uncertainty, which is used in US and QBC to query the most uncertain samples, is related to human uncertainty. Another interesting insight is, that even with a worse mean labeling quality, using US and QBC resulted in a better performing classifier than RAND (see Fig. 5).

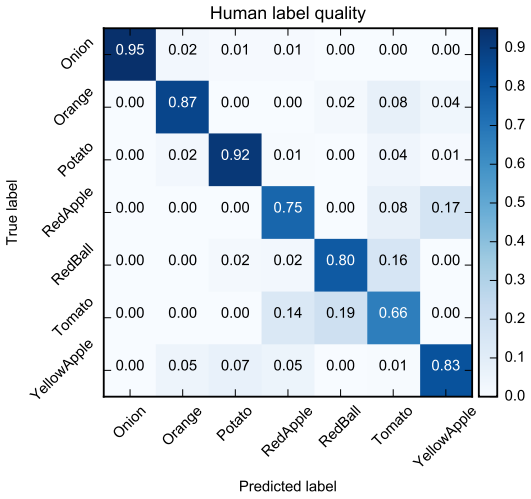

Fig. 6: Confusion matrix of human labels from all compared querying approaches.

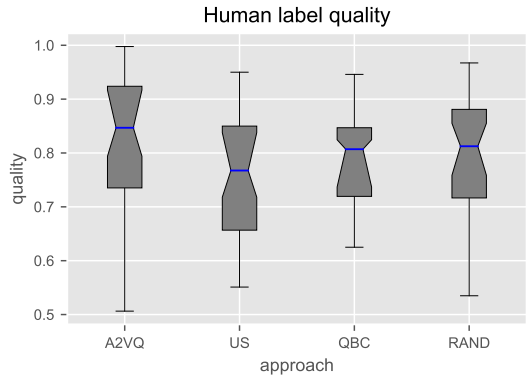

Fig. 7: Human label quality for tested approaches.

Figure 8 shows how many samples were labeled within five minutes in the different experimental trials. The figure indicates, that people could label more samples using A2VQ while the number of labeled samples of the baseline approaches were comparable. The result of the statistical tests confirmed this observations. This outcome is as expected, because with A2VQ people can label multiple images with the same label while in baseline approaches just one image can be labeled at a time.

6 Conclusion

In this paper we have proposed to use dimension reduction techniques for applying active learning with a visualization. Therefore we introduced the querying approach A2VQ which queries optimal views for labeling by the user. We developed a user interface which implements A2VQ and was also evaluated in a user study. For the used OUTDOOR data set, the study showed that using A2VQ improves the classifier's accuracy, the number of labeled samples and also the label quality compared to US, QBC and random querying.

There are some possible directions for interesting further research in this field. The user study showed that baseline methods have the advantage to faster respond at the start of training. When training samples can be ambiguous, we showed that the used DQBE [11] approach has a huge impact in boosting the speed by querying only meaningful samples. However, our study showed that after 100 seconds the fast increase in accuracy of the baseline methods saturates. So it may be worth to evaluate a hybrid model, that first uses a baseline technique to query a few samples of each class for the fast training of an initial classifier. Following this, A2VQ could be used to label in depth. Using A2VQ also results in a higher label quality, as our study showed. Therefore, it may also correct former contradictions in labels, since we think that seeing patterns in contrast to other patterns facilitate to give the correct label.

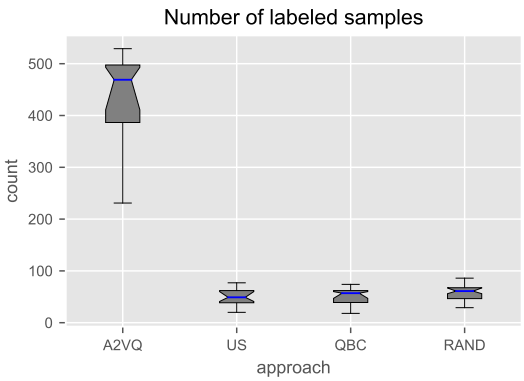

Fig. 8: Number of labeled samples of the different approaches.

It may be possible to use semi-supervised dimension reduction techniques [20] for a better visualization. Doing so, after each trained view not only the classifier is retrained, but also the visualization is regenerated with new label information.

In the near future we will integrate A2VQ together with the labeling interface within a service robot [2], which interacts in a smart lobby environment. By showing the user interface on the robot's front touch screen we want to allow the user not only to teach the robot objects by a finger swipe, but also give him a feeling what the robot's internal representation of the objects might be.

References

1. Cavallo, M., Demiralp, Ç.: A visual interaction framework for dimensionality reduction based data exploration. In: Conference on Human Factors in Computing Systems CHI. p. 635 (2018)

2. Hasler, S., Kreger, J., Bauer-Wersing, U.: Interactive incremental online learning of objects onboard of a cooperative autonomous mobile robot. In: International Conference on Neural Information Processing. pp. 279–290 (2018)
3. Hinton, G.E., Roweis, S.T.: Stochastic neighbor embedding. In: Advances in neural information processing systems (NIPS). pp. 857–864 (2003)
4. Hotelling, H.: Analysis of a complex of statistical variables into principal components. Journal of educational psychology **24**(6), 417 (1933)
5. Huang, L., Matwin, S., de Carvalho, E.J., Minghim, R.: Active learning with visualization for text data. In: ACM Workshop on Exploratory Search and Interactive Data Analytics. pp. 69–74 (2017)
6. Iwata, T., Houlsby, N., Ghahramani, Z.: Active learning for interactive visualization. In: International Conference on Artificial Intelligence and Statistics AISTATS. pp. 342–350 (2013)
7. Kohonen, T.: Self-organized formation of topologically correct feature maps. Biological cybernetics **43**(1), 59–69 (1982)
8. Lewis, D.D., Gale, W.A.: A sequential algorithm for training text classifiers. In: International Conference on Research and Development in Information Retrieval (ACM-SIGIR). pp. 3–12 (1994)
9. Liao, H., Chen, L., Song, Y., Ming, H.: Visualization-based active learning for video annotation. IEEE Trans. Multimedia **18**(11), 2196–2205 (2016)
10. Limberg, C., Wersing, H., Ritter, H.: Efficient accuracy estimation for instance-based incremental active learning. In: European Symposium on Artificial Neural Networks (ESANN). pp. 171–176 (2018)
11. Limberg, C., Wersing, H., Ritter, H.: Improving active learning by avoiding ambiguous samples. In: International Conference on Artificial Neural Networks (ICANN). Springer (October 2018)
12. Losing, V., Hammer, B., Wersing, H.: Interactive online learning for obstacle classification on a mobile robot. In: International Joint Conference on Neural Networks (IJCNN). pp. 1–8 (2015)
13. Maaten, L.v.d., Hinton, G.: Visualizing data using t-sne. Journal of machine learning research **9**, 2579–2605 (2008)
14. Ramirez-Loaiza, M.E., Sharma, M., Kumar, G., Bilgic, M.: Active learning: an empirical study of common baselines. Data Mining and Knowledge Discovery **31**(2), 287–313 (2017)
15. Settles, B.: Active learning. Synthesis Lectures on Artificial Intelligence and Machine Learning **6**(1), 1–114 (2012)
16. Settles, B., Craven, M.: An analysis of active learning strategies for sequence labeling tasks. In: Conference on Empirical Methods in Natural Language Processing (EMNLP). pp. 1070–1079 (2008)
17. Seung, H.S., Opper, M., Sompolinsky, H.: Query by committee. In: Conference on Computational Learning Theory (COLT). pp. 287–294 (1992)
18. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. CoRR **abs/1409.1556** (2014)
19. Torgerson, W.S.: Multidimensional scaling: I. theory and method. Psychometrika **17**(4), 401–419 (1952)
20. Zhang, D., Zhou, Z., Chen, S.: Semi-supervised dimensionality reduction. In: SIAM International Conference on Data Mining. pp. 629–634 (2007)