Automated Selection of High-Quality Synthetic Images for Data-Driven Machine Learning: A Study on Traffic Signs

Daniela Horn* Lars Janssen* Sebastian Houben*

Abstract— **The utilization of automatically generated image training data is a feasible way to enhance existing datasets, e.g., by strengthening underrepresented classes or by adding new lighting or weather conditions for more variety. Synthetic images can also be used to introduce entirely new classes to a given dataset. In order to maximize the positive effects of generated image data on classifier training and reduce the possible downsides of potentially problematic image samples, an automatic quality assessment of each generated image seems sensible for overall quality enhancement of the training set and, thus, of the resulting classifier.**

In this paper we extend our previous work on synthetic traffic sign images by assessing the quality of a fully generated dataset consisting of 215,000 traffic sign images using four different measures. According to each sample's quality, we successively reduce the size of our training set and evaluate the performance with SVM and CNN classifiers to verify the approach. The comparability of real-world and synthetic training data is investigated by contrasting several classifiers trained on generated data to our baseline w.r.t. actual misclassifications during testing.

I. Introduction

Many state-of-the-art forms of machine learning are highly data-dependent and new, previously unheard of highs in performance rely in part on huge data acquisition and labeling projects that try to gather and structure the required amounts of training and test data, e.g., [1]–[4]. This demand has driven the development of both generative machine learning (ML) models and close-to-life simulations, either of which holds the prospect of superseding this cumbersome task in the future [5] and, in easier cases with well-known sources of variance, even today [6]. Apart from fast, cost-efficient, and life-like data generation, the opportunity of systematic data coverage instead of real-world, hence randomly sampled, data acquisition is perceived as a clear advantage. This applies for both model training and the systematic evaluation of these usually safety-critical systems.

Unlike human-defined simulations, ML-powered generation techniques oftentimes lack a straightforward possibility to control or manipulate the result and, thus, the distribution of the generated data. This is why a number of hybrid approaches have tried to combine the strengths of ML models and simulations [7], [8]. They aim to generate part of the distribution by learning from real data and the other part by using hard-coded rules, e.g., provided by a human programmer. However, both purely data-driven and hybrid models suffer from outliers and failure cases in which the resulting data points are invalid or highly unlikely when compared to the targeted real-world distribution.

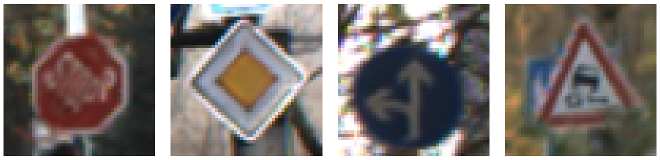


Fig. 1. How useful are these generated training images for later classification of real-world traffic signs? Human perception can be deceiving when it comes to assessing well- or ill-suited images for the training of a machine learning algorithm. The two images on the left received the highest score with our ACC measure and lead to a more robust classifier. The two images on the right were given the lowest possible rating with the same measure and the resulting classifier shows that they are actually harmful to its performance.

In order to assess the quality of the generated data, both human acceptance studies and various objective metrics have been proposed and tested in practice [9], [10]. While human beings might evaluate individual samples one way or the other for wrong reasons (cf. Fig. 1), their judgement is subjective and based on human perception, which can be misleading when dealing with ML approaches. Still, many objective metrics compare the generated and real-world distribution as a whole, ignoring the possible impact of individual image samples.

In this paper, we propose and investigate several automated methods for assessing the quality of a single generated data point. We study these approaches on our recently proposed system for traffic sign substitution which uses real-world images of traffic signs and generates a corresponding synthetic image by exchanging the class identity. The entire hybrid approach consists of several steps and provides a number of failure modes in both the rule-based and the ML-driven part.

We identify those failure modes and introduce four quality measures to automatically overcome prior problems, reduce the size of a given training dataset, and enhance the overall quality at the same time. The resulting datasets are evaluated via accuracy of the classifiers trained on them and by comparison of misclassified test images with those falsely classified by a baseline classifier, which was trained on the training dataset of the German Traffic Sign Recognition Benchmark (GTSRB) [4].

II. Related Work

For a brief overview of the image generation system that is the foundation of this paper [11], [12], we refer the reader to Sec. III-A. Other parts of our approach draw from several

* The authors are with the Institute for Neural Computation, Ruhr University Bochum, Universitaetsstrasse 150, 44780 Bochum, Germany `firstname.lastname@ini.rub.de`

active fields of research for which the related references are pointed out below:

In the last few years, CycleGANs, as originally proposed by Zhu et al. [13], have been used for diverse generation problems. A major asset of CycleGANs is the amount of control that can be gained over the generation process if complex image data is transferred to a simple and easy-to-manipulate domain. In combination with the fact that they can be trained with unpaired datsets, CycleGANs have become more and more attractive for a variety of use cases. In 2020, Liu et al. [14] adopted a CycleGAN architecture for a day-to-night style transfer in order to amend a highly complex traffic scene dataset with an immense shortage of nighttime scenes. In the same year, Mălăescu et al. [15] presented a CycleGAN-based approach to transfer obtained training data from a low-resolution driver surveillance camera to resemble data that were obtained with a newer high-resolution camera model.

During the past decade, the task of image-based traffic sign recognition has been subject of several research projects. A number of datasets were published covering traffic signs of different countries [4], [16]–[18], but the variety of traffic sign locations makes them hardly sufficient to train robust classifiers. The variance in lighting and weather conditions as well as smaller changes in coloring and style leave traffic sign recognition to be a challenging problem but at the same time a prime example for image data generation since the geometry is simple and perspective changes can be computed straightforwardly. Luo et al. [19] have demonstrated this early on by a style transfer and randomly cropping background information. Our own methods aim to create realistic backgrounds from scratch [11] or purely reuse those background and recording conditions that have been featured in the underlying real-world dataset [12] to allow more realism and fewer necessary training examples.

The approach in this paper is built around different methods for uncertainty estimation. These intend to enable a trained model to recognize when a given input is dissimilar to other examples from its original training distribution. We cite the survey by Ovadia et al. [20] for a concise introduction into the field.

III. Method

As this paper extends our earlier work, we briefly introduce our generation pipeline for synthetic traffic sign samples in Sec. III-A. In Sec. III-B, we identify remaining failure modes of the generation process before elaborating on the measures with which we rate the quality of individual image samples to enhance a given dataset in Sec. III-C.

A. Previous Work

The generative pipeline at the center of this study consists of a CycleGAN architecture as its core mechanism which performs a style transfer between life-like images and cartoon representations of traffic signs. The CycleGAN was trained on the GTSRB and an unpaired set of cartoon images featuring a traffic sign icon in a random but life-like pose and a homogeneously colored background [11].

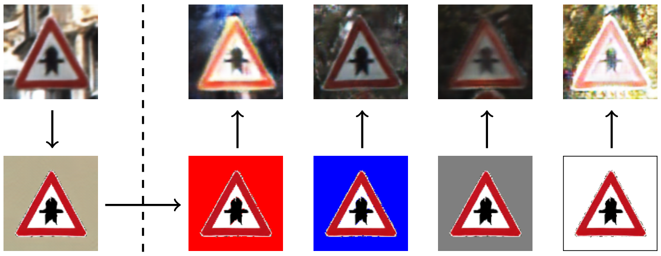


Fig. 2. Overview of the general CycleGAN approach. A real-world sample (*top left*) is transferred to its cartoon representation (*bottom left*). Information on background, illumination and blurriness of the original is encoded in the unicolored background of the cartoon sample. By changing the background color in the cartoon domain and transferring it to the life-like domain, the CycleGAN creates new surroundings for the given traffic sign. (Figure as seen in [12])

As expressiveness of the CycleGAN is limited, it often fails to create a real-life background from a uniformly colored one (cf. Fig. 2). For this reason, we pursue a substitution approach [12] in which a real-life traffic sign is replaced by one of the same category but not necessarily the same class. To this end, the original image is transformed into its cartoon pendant. This representation provides an almost homogeneous background but encodes background structure in minuscule variations. The pose in this simplified version is estimated by fitting an icon of the original class and replacing it with an icon of the target class transformed to the same pose. Finally, performing a style transfer into the real-life domain retains the background from before but replaces the traffic sign with the selected one.

B. Failure Modes

The presented system suffers from three main failure modes (cf. Fig. 3) that do not occur particularly frequently but prohibit fully automated use:

1) *Pose estimation fails, usually due to inaccurate keypoint matching*: This may happen by cause of an imperfect style transfer from real-world to cartoon domain. Other reasons are traffic signs that display uncommon iconography or fonts that differ from the ones used on the fitted template.
2) *Style transfer fails*: As the CycleGAN was trained under the side constraint of retaining class identity, in rare cases icons are added to or inpainted in the transferred cartoon image. This can also happen if the image quality is insufficient or the background holds a similar color as the sign.
3) *Embedding of the substituted sign is not life-like*: The entire approach builds on the assumption that important information for a realistic embedding is encoded in the background of the cartoon images, e.g., lighting, weather, and contrast. At times, however, the manipulated cartoon image does not correspond to a life-like image in which foreground and background agree.

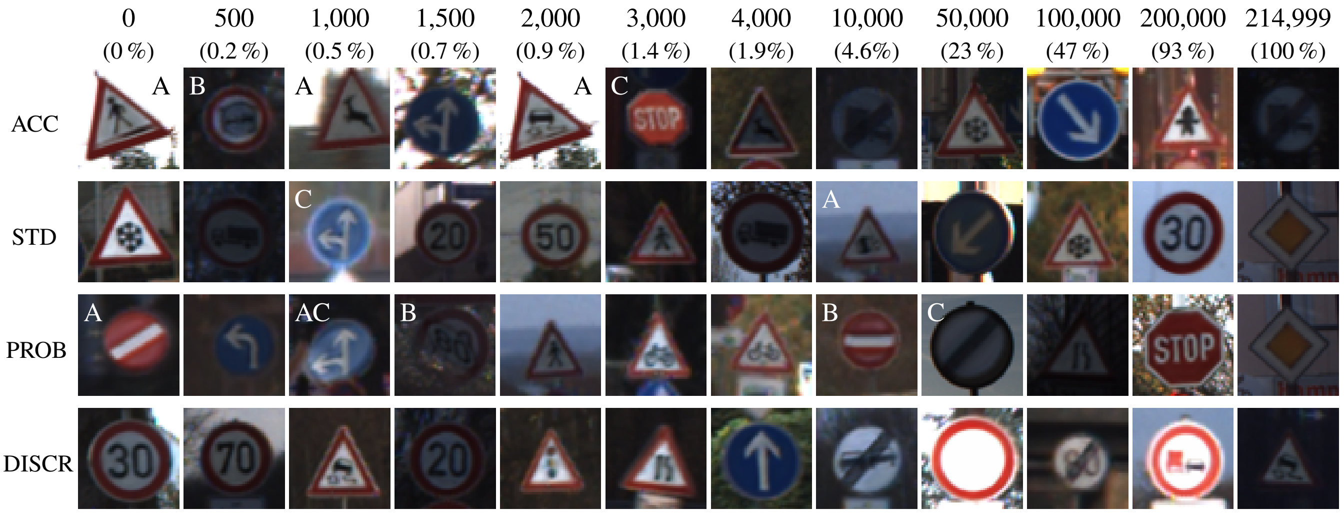


Fig. 3. Ranking examples according to chosen quality method: All images from our data generation pipeline are arranged according to each of the proposed quality metrics. The column heading states the absolute and relative position of the respective example within the particular arrangement (0 being the one with lowest and $214,999$ the one with highest quality evaluation). Examples with prominent failure modes during substitution are pointed out: (A) unusual poses or failure of pose estimation, (B) failure of style transfer, (C) chosen style does not fit surrounding. Please note that for the reduction of the training sets as described in Fig. 4 selection was performed for each class individually, hence, balancing the number of examples for all classes. This overview disregards class information.

C. Quality Measures for Synthesized Images

In order to measure the quality of generated images we construct four metrics that each operate on single data points, i.e., individual image samples. Three of these are based on the uncertainty estimation resulting from the use of Monte Carlo (MC) Dropout, as described in [21]. With this method, all Dropout layers of a neural network, i.e., layers in which neurons are randomly set to zero output, are activated not only at training but also at testing time to provide a statistic on the network's predictions. To estimate a measure of the uncertainty of an image, the result of T forward passes through the network is averaged.

As there are different methods to interpret the output we construct three quality metrics: *PROB*, *STD*, and *ACC*. A CNN classifier trained on the training dataset of the GTSRB consisting of only real images serves as our model. By estimating the uncertainty of generated images using the real-trained classifier, unrealistic synthetic images may be identified.

Initially, the model's output, i.e., the softmax vector over 43 traffic sign classes as defined by the GTSRB dataset, is averaged over all T forward passes. The highest probability then gives the value of the PROB metric. In doing so, a large value is interpreted as the model being certain about its prediction regardless of whether or not it corresponds to the correct class. A low value therefore indicates low output of all classes and thus an uncertain decision.

The STD metric is composed by determining the standard deviation over all T passes of the class with the highest softmax output. A large value results from a large variation in the output and thus is interpreted as an uncertain classification. Again, the correctness of the classification is of no importance to the results of this measure.

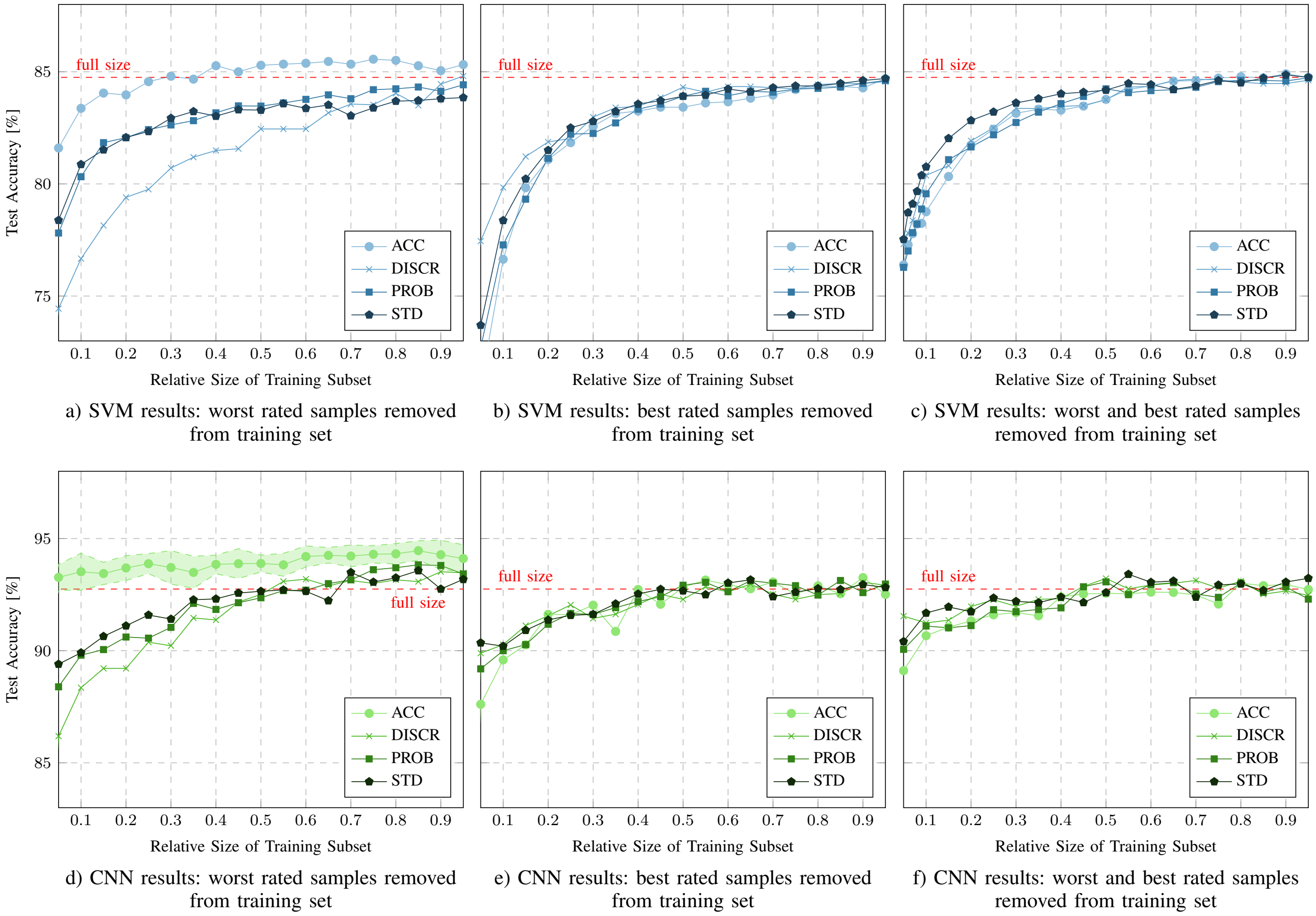
As we have not taken into account if the predicted class is actually the correct one, we construct the third metric, which we denote by ACC, using an accuracy score by checking if the correct class has the highest output in each of the T runs. A high accuracy thus may represent the presence of many realistic properties in the image as the classifier is able to "easily" classify it.

The fourth quality metric, the *DISCR* metric, is not based on a measure of uncertainty but uses the discriminator of the above introduced CycleGAN (cf. Sec. III-A), which was trained to tell apart real-world images from those transferred from cartoon to real-life domain by the corresponding generator network. The discriminator, thus, assesses the realism of the given image.

IV. Experiments

For all experiments we use the generated images of the same dataset for training. It consists of 43 different traffic sign classes with $5,000$ samples per class, adding up to a dataset size of $215,000$ in total. The dataset was adopted from our earlier work [12], however, all generated images were scaled from the original 128×128 to a resolution of 48×48. We found this to be in line with other experiments in the literature [16] and, while matching the previous performance, classifier training routine speeds up considerably.

In order to evaluate the quality of the four measures introduced in Sec. III-C, we use their assessments of each image sample from the training dataset to prune it in a predefined way. In three series of experiments, we reduce the dataset size by removing the worst rated images, the best rated ones, or images from both ends of the spectrum. This is done in steps of 5 percentage points for each metric, resulting in relative dataset sizes of $95\,\%$ to $5\,\%$.



a) SVM results: worst rated samples removed from training set

b) SVM results: best rated samples removed from training set

c) SVM results: worst and best rated samples removed from training set

d) CNN results: worst rated samples removed from training set

e) CNN results: best rated samples removed from training set

f) CNN results: worst and best rated samples removed from training set

Fig. 4. SVM and CNN results for partial datasets. *From left to right:* Dataset reduction by removal of worst rated samples, removal of best rated samples, and symmetric removal of worst and best rated samples w.r.t. a certain measure. CNN results were averaged over 10 runs for each measure and dataset size. Plot d) shows the standard deviation of the CNN classifiers for the ACC measure for better understanding. All experiments feature the same spread of around $\pm 1\,\%$ which is not shown for reasons of readability. The red dashed line depicts the respective classifier performance trained with the full training set.

While the removal of the worst rated images is a rather intuitive approach to eliminate weak or even unfavorable data from the training set, the elimination of the best images follows another train of thought: Generated images can raise difficulties if they are too immaculate, as they have lost the noise and imperfections of real-world samples. Thus, they might be perfectly classified and therefore highly rated by our measures but ultimately lead to a weaker classifier for the real world as the overall distribution of images loses variance. In order to investigate the impact of too perfectly generated samples, we have chosen the dataset size reduction by removal of the best rated samples, as well. The third series of experiments combines the concepts and thus strengths of the first two sets.

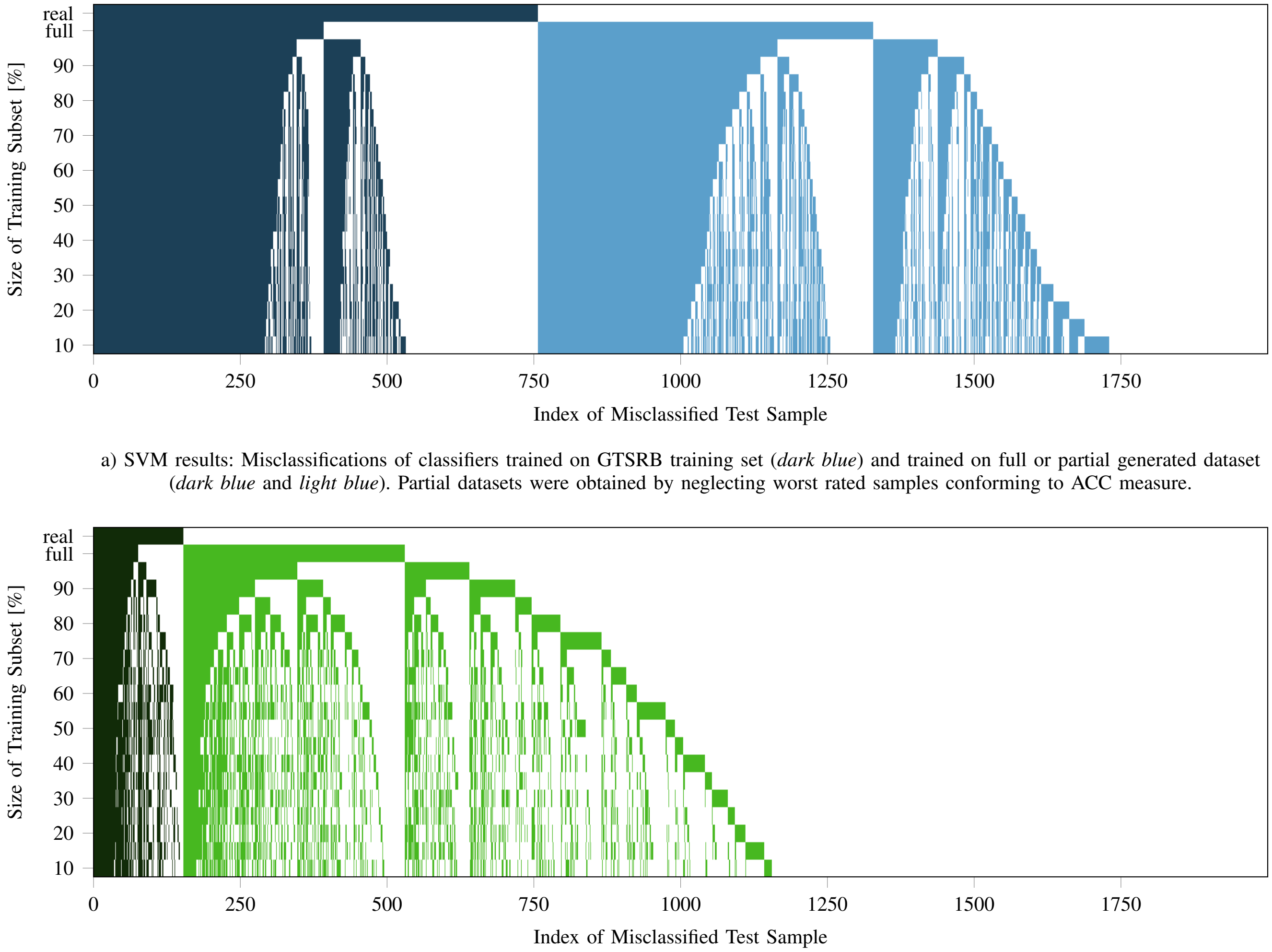
In order to maintain balanced classes for each reduced dataset, pruning has to be class-wise. Without this restriction, more complex traffic sign classes might be underrepresented or even vanish completely from the training set.

Each experiment is conducted with both CNN classifiers and SVM classifiers based on HOG features. The SVM and CNN specifications resemble those used in [11], however, the CNN classifiers were given an adaptive learning rate, depending on the size of the training dataset, in order to achieve invariance of the dataset size. As the training process of CNNs is heavily initialization dependent and thus their performance can vary remarkably, the displayed results were averaged over 10 runs each.

We use half of the original GTSRB test set for all experiments, as specified in [11], amounting to $6,315$ real-world images. As with the training data, we resize all test samples to 48×48 for the CNN classifiers, but maintain the various given GTSRB sample sizes for all SVM classifiers.

V. Results

The described experiments give rise to manifold interpretations: In Sec. V-A the focus is set to the overall quality of the partial training sets reduced according to given measures and in three different ways. While the performance of the resulting classifiers is the main interest in training data generation, another question is the proximity of the generated data to the real-world reference training set, i.e., the GTSRB training set. In Sec. V-B misclassified test images of several reduced datasets are compared to the baseline. The higher



a) SVM results: Misclassifications of classifiers trained on GTSRB training set (*dark blue*) and trained on full or partial generated dataset (*dark blue* and *light blue*). Partial datasets were obtained by neglecting worst rated samples conforming to ACC measure.

b) CNN results: Misclassifications of classifiers trained on GTSRB training set (*dark green*) and trained on full or partial generated dataset (*dark green* and *light green*). Partial datasets were obtained by neglecting worst rated samples conforming to ACC measure.

Fig. 5. Misclassifications of SVM and CNN classifiers trained on partial datasets created in compliance with ACC measure. Rows denote 5 % steps in dataset size reduction, with classifiers trained on GTSRB training set and full generated dataset on top for direct reference. Darker shades refer to misclassifications by classifiers trained on real data, lighter shades to misclassifications only committed by classifiers trained on generated data.

the overlap of misclassified samples, the closer the generated data to the real-world dataset.

A. *Performance of Measures*

The performance of SVM and CNN classifiers for partial datasets is depicted in Sec. 4. The dataset sizes range from 95 % to 5 %; the performance of the full sized generated dataset is given by a red dashed line. Plot d) additionally shows the exemplary standard deviation of 10 CNN runs for the ACC measure. For all experiments this spread is around ±1 % and is therefore disregarded.

Both classifier types show a significant increase in test performance for ACC when removing the worst rated samples from the dataset (cf. Fig. 4a,d). In this case, the overall dataset size could be reduced to 40 % for the SVMs and to a mere 5 % for the CNNs without losing accuracy. While no other measure profits from a dataset reduction for the SVMs, the CNNs result in a performance increase for all measures when reducing the size by up to 30 %. CNNs, thus, seem to be easily confused by badly generated samples, SVMs based on HOG features, in comparison, show a more robust behavior.

For the other two reduction approaches, all measures show similar performance within a classifier and reduction type and results do not increase significantly over the full sized dataset. As the ACC measure in combination with the removal of worst rated samples clearly outperforms any other measure or reduction technique for both classifier types, this setup is investigated further w.r.t. proximity of misclassifications to the baseline.

B. *Comparison of Misclassifications*

A side product of the experiments described above are lists of misclassified images for each single setup. In order to gain a deeper understanding of the failure case composition and thus the comparability of our generated training datasets to the baseline, the distributions of these misclassifications are compared as well. Fig. 5 shows the respective results for the

SVM and CNN classifiers produced by the ACC measure. For CNN results out of 10 available runs per experiment a random one was chosen for display.

Each row depicts the misclassified training samples of a certain classifier, with the baseline always on top, followed by a descending order w.r.t. dataset size for the classifiers trained on generated images. Each horizontal bar reduces the respective dataset size by 5 percentage points. The two colors divide misclassifications w.r.t. the performance of the baseline classifier. If a test sample was misclassified by the baseline classifier, it is depicted in a darker shade. Was the sample classified correctly in the baseline case but misclassified in any of the generated training set cases, a lighter color has been used to clearly mark the difference. This type of plot displays how the sets of misclassified examples overlap among classifiers, i.e., whether reducing the training set size results in substantially the same or significantly different misclassifications. Furthermore one might regard misclassifications by the baseline classifier as forgivable and investigate the classification for these examples in all other deployed models.

A direct comparison of both plots shows that while the misclassifications of the CNNs are less in number their occurrence is rather scattered both w.r.t. the real-world training data and other generated datasets. They seem to falsely recognize different images with every new run, resulting in a noisy graph. The SVMs on the other hand give a more unison impression of misclassifications. These classifiers trained on different amounts of the same generated dataset show a tendency to misclassify the same test samples and also the comparison to the baseline depicts a certain concordance of test sample (mis-)understandings.

VI. Discussion and Future Work

For all proposed metrics, we use a surrogate task to decide whether or not a given image is significantly different from the distribution of real images. The three approaches based on MC Dropout aim to take advantage of the fact that unrealistic images may bear features of several of the possible classes which in turn should impede robust classification. The DISCR metric is trained to tell fake and real images apart but is tuned to the most discriminative, i.e., most unrealistic, features of the particular generator with which it had been paired.

We have shown that the ACC quality measure provides a useful ranking of generated images: It allows the selection of images with failure cases without reducing the overall variance in the training set overmuch. Removing a percentage of the lowest-ranked images leads to an increase in performance and enables the reduction of the necessary training set size. However, since ACC is based on a measure of spread in traffic sign classes present in the real-world dataset, it remains to be investigated whether generating entirely new classes, i.e., classes with new symbols that the original dataset lacks and the base classifier has therefore not seen, will bias the quality measure.

References

[1] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "Imagenet: A Large-Scale Hierarchical Image Database," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2009, pp. 248–255.

[2] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick, "Microsoft COCO: Common Objects in Context," in *European Conference on Computer Vision*, 2014, pp. 740–755.

[3] S. Houben, J. Stallkamp, J. Salmen, M. Schlipsing, and C. Igel, "Detection of Traffic Signs in Real-World Images: The German Traffic Sign Detection Benchmark," in *IEEE International Joint Conference on Neural Networks*, 2013, pp. 714–721.

[4] J. Stallkamp, M. Schlipsing, J. Salmen, and C. Igel, "The German Traffic Sign Recognition Benchmark: A Multi-Class Classification Competition," in *IEEE International Joint Conference on Neural Networks*, 2011, pp. 1453–1460.

[5] T.-C. Wang, M.-Y. Liu, A. Tao, G. Liu, J. Kautz, and B. Catanzaro, "Few-Shot Video-to-Video Synthesis," in *Advances in Neural Information Processing Systems*, 2019, pp. 5013–5024.

[6] M.-Y. Liu, T. Breuel, and J. Kautz, "Unsupervised Image-to-Image Translation Networks," in *Advances in Neural Information Processing Systems*, 2017, pp. 700–708.

[7] X. Chen, Y. Duan, R. Houthooft, J. Schulman, I. Sutskever, and P. Abbeel, "InfoGAN: Interpretable Representation Learning by Information Maximizing Generative Adversarial Nets," in *Advances in Neural Information Processing Systems*, 2016, pp. 2172–2180.

[8] A. Odena, C. Olah, and J. Shlens, "Conditional Image Synthesis with Auxiliary Classifier GANs," in *International Conference on Machine Learning*, 2017, pp. 2642–2651.

[9] J. Frank, T. Eisenhofer, L. Schönherr, A. Fischer, D. Kolossa, and T. Holz, "Leveraging Frequency Analysis for Deep Fake Image Recognition," in *International Conference on Machine Learning*, 2020, pp. 3247–3258.

[10] A. Borji, "Pros and Cons of GAN Evaluation Measures," *arXiv:1802.03446 [cs.CV]*, 2018.

[11] D. Spata, D. Horn, and S. Houben, "Generation of Natural Traffic Sign Images Using Domain Translation with Cycle-Consistent Generative Adversarial Networks," in *IEEE Intelligent Vehicles Symposium*, 2019, pp. 622–628.

[12] D. Horn and S. Houben, "Fully Automated Traffic Sign Substitution in Real-World Images for Large-Scale Data Augmentation," in *IEEE Intelligent Vehicles Symposium*, 2020, pp. 194–200.

[13] J.-Y. Zhu, T. Park, P. Isola, and A. A. Efros, "Unpaired Image-to-Image Translation using Cycle-Consistent Adversarial Networks," in *IEEE International Conference on Computer Vision*, 2017, pp. 2223–2232.

[14] T. Liu, Z. Chen, Y. Yang, Z. Wu, and H. Li, "Lane Detection in Low-Light Conditions Using an Efficient Data Enhancement: Light Conditions Style Transfer," in *IEEE Intelligent Vehicles Symposium*, 2020, pp. 1123–1128.

[15] A. Mălăescu, A. Frăţilă, L. C. Duţu, A. Sultana, D. Filip, and M. Ciuc, "Task-Driven Image-to-Image Translation for Automotive Applications," in *IEEE Intelligent Vehicles Symposium*, 2020, pp. 1855–1861.

[16] C. Gámez Serna and Y. Ruichek, "Classification of Traffic Signs: The European Dataset," *IEEE Access*, vol. 6, pp. 78 136–78 148, 2018.

[17] R. Timofte, M. Mathias, R. Benenson, and L. Van Gool, "Traffic Sign Recognition - How far are we from the solution?" in *IEEE International Joint Conference on Neural Networks*, 2013, pp. 1–8.

[18] F. Larsson and M. Felsberg, "Using Fourier Descriptors and Spatial Models for Traffic Sign Recognition," in *Scandinavian Conference on Image Analysis*, 2011, pp. 238–249.

[19] H. Luo, Q. Kong, and F. Wu, "Traffic Sign Image Synthesis with Generative Adversarial Networks," in *IEEE International Conference on Pattern Recognition*, 2018, pp. 2540–2545.

[20] Y. Ovadia, E. Fertig, J. Ren, Z. Nado, D. Sculley, S. Nowozin, J. V. Dillon, B. Lakshminarayanan, and J. Snoek, "Can You Trust Your Model's Uncertainty? Evaluating Predictive Uncertainty Under Dataset Shift," in *Advances in Neural Information Processing Systems*, 2019, pp. 13 991–14 002.

[21] Y. Gal and Z. Ghahramani, "Dropout as a Bayesian Approximation: Representing Model Uncertainty in Deep Learning," in *International Conference on Machine Learning*, 2016, pp. 1050–1059.