# Data augmentation with mixtures of max-entropy transformations for filling-level classification

Apostolos Modas
*EPFL*
Switzerland
apostolos.modas@epfl.ch

Andrea Cavallaro
*Queen Mary University of London*
United Kingdom
a.cavallaro@qmul.ac.uk

Pascal Frossard
*EPFL*
Switzerland
pascal.frossard@epfl.ch

*Abstract*—We address the problem of distribution shifts in test-time data with a principled data augmentation scheme for the task of content-level classification. In such a task, properties such as shape or transparency of test-time containers (cup or drinking glass) may differ from those represented in the training data. Dealing with such distribution shifts using standard augmentation schemes is challenging and transforming the training images to cover the properties of the test-time instances requires sophisticated image manipulations. We therefore generate diverse augmentations using a family of max-entropy transformations that create samples with new shapes, colors and spectral characteristics. We show that such a principled augmentation scheme, alone, can replace current approaches that use transfer learning or can be used in combination with transfer learning to improve its performance.

*Index Terms*—data augmentation, transfer learning, filling level estimation

## I. INTRODUCTION

In many machine learning tasks, transfer learning [1] helps overcome the limited amount of training data [2]–[7]. However, pre-training models on large datasets designed for another task gives no guarantees on the relevance of the transferred features for the target task. An alternative to increase training-data variability is with data augmentation [8]. However, generating new training samples that are representative of test-time data might require complex operations, including composition of transformations or mixing strategies [9].

Robot perception offers several examples of real-world machine learning tasks that rely on scarce training data. One of such examples is human-robot interaction, where estimating the physical properties of objects is important for executing safe grasps [10]. An important property to be estimated is the weight of the object (e.g. a container with unknown content) through computer vision, which requires the inference of the shape of the container [11], and the type and amount of its content [12], [13]. Classifying the filling level of an *unknown* container, though, is challenging due to test-time distribution shifts caused by hand occlusions, transparencies of both the container and the filling [6], and shape differences among containers. Current approaches use RGB [12], thermal [14], or a combination of RGB and depth data [15], [16] and cannot rely on large task-specific training data. Therefore, transfer learning with classifiers pre-trained on large datasets (i.e. ImageNet [17]) is typically used [12], [13] instead of
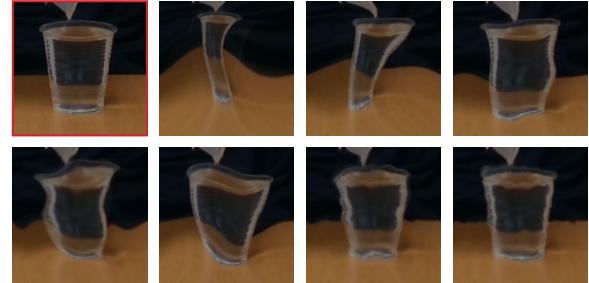


Fig. 1. Sample transformed images using diffeomorphisms with varying smoothness level $K_\tau$ (top row, from left to right: original image, transformed image with $K_\tau = 2, 5, 10$; bottom row: $K_\tau = 20, 40, 100, 300$).

data augmentation. Although data augmentation imposes explicitly the changes that the classifiers should be invariant to (e.g. random noise, random horizontal flips), typical data-augmentation transformations are limited in expressing more realistic distribution shifts, such as common corruptions [18]. More complex operations that are based on compositions of transformations would be preferable [9].

In this work, we address the filling-level classification training problem using data augmentation with a family of max-entropy transformations, along with a mixing strategy [19]. These transformations allow us to generate diverse and targeted augmentations that can be tuned to generate samples tailored to train classifiers that generalize on containers with shapes (Fig. 1), color, and spectral content that are not represented in the original training data. This constructive approach for augmentation leads to a filling-level classification accuracy that is on-par with, or better than, the accuracy of transfer learning [13], but with a much smaller training dataset. This augmentation scheme requires only $1.2\times$ additional training time compared to standard training on a small dataset suitable for estimating the filling level ($\sim$10 thousand images) and is computationally much less expensive than pre-training on ImageNet ($\sim$1.2 million images). We also show that the performance of the classifier may further increase when this data augmentation scheme is used in concert with transfer learning itself [13].

## II. AUGMENTATIONS FOR FILLING-LEVEL CLASSIFICATION

We approach the problem of estimating the filling level, $y$, of a container in an image $\boldsymbol{x}$, as a classification

task [12], [13]. We express the filling level as $y \in \{empty, half\text{-}full, full, unknown\}$, where the *unknown* class covers cases where the filling level cannot be estimated directly (e.g. opaque containers). We curated C-CCM [13], a subset of the CORSMAL Containers Manipulation dataset [20] with images of cups and glasses that may contain water, pasta or rice[1]. The shape and transparencies of the containers, which are captured under different illumination conditions, background types and occlusion levels, vary substantially. We defined three training and validation splits [13], where a distribution shift is introduced in the validation set, such that some containers always have a property that does not exist in the training set (e.g. a unique shape or color).

### A. Adversarial transfer learning

With standard training (ST), the classifier learns directly on the available data for the target task. For small datasets, this typically results in classifiers with low accuracy on the test set due to overfitting [12], [13]. Overfitting is caused by the over-reliance of the network on specific, possibly spurious, features that maximize the training accuracy, but do not capture the variability of the categories of interest at test time.

Transfer learning helps preventing the network from overfitting, by first performing ST on a large dataset representing the *source domain* (e.g. ImageNet), and then fine-tuning (FT) the resulting network on a smaller dataset representing the *target domain* (e.g. C-CCM for filling-level classification). We denote standard transfer learning as ST→FT.

The performance of transfer learning for filling-level classification further improves if the network is first adversarially trained in the source domain [13]. Adversarial training (AT) [21] is a data augmentation technique that replaces, during training, the original images with their adversarial examples [22], [23]. We denote this transfer learning approach that uses adversarial learning in the source domain as AT→FT.

Transfer learning, and especially AT→FT, improves the filling-level classification performance [13]. However, applying adversarial training on a large dataset such as ImageNet is extremely expensive. Moreover, exploiting information from datasets like ImageNet might not always relate to the actual information required by the target task. By transferring this knowledge to the new task, there is an expectation (i.e. not an explicit design choice) that the target network will learn more generalizable features. However, there is yet no mechanism for transfer learning to specify explicitly which features of the training data the target network should be more responsive to.

To avoid the computational cost of training robust models with transfer learning we investigate how to enrich the training set with data augmentations that effectively increase the variability of data with interpretable image modifications, as discussed in the next section.

### B. Principled data augmentation

The performance on the validation set of C-CMM on the "shifted" containers is systematically lower than that on containers that share similarities with those in the training set (overfitting) [13]. To address this limitation, we consider a data augmentation scheme that generates diverse augmentations using a set of primitive max-entropy transformations on the spatial $\tau$, color, $\gamma$, and spectral, $\omega$, domain [19]. We expect these transformations to relate to the changes we want to introduce during training: container shape through $\tau$, container color through $\gamma$, and illumination and texture through $\omega$. Each transformation requires only two control parameters, namely one for the smoothness, $K$, and one for the strength, $\sigma^2$.

A transformed image $\boldsymbol{x}_T$ is generated through a convex combination of $n$ basic augmentations (width) consisting of the composition of $m$ of its max-entropy transformations (depth) [19]:

$$\boldsymbol{x}_T = \sum_{i=1}^{n} \mu_i g_1^i \circ \cdots \circ g_m^i(\boldsymbol{x}) : g \in \{\tau, \gamma, \omega\}, \quad (1)$$

where $\boldsymbol{\mu} \sim$ Dirichlet distribution. The width $n$ specifies the number of transformed instances to be used in the convex combination for synthesizing the transformed image $\boldsymbol{x}_T$, while the depth $m$ specifies how many transformations will be sequentially applied on an image.

The final image $\hat{\boldsymbol{x}}$ is synthesized as a linear combination of the original image $\boldsymbol{x}$ and the result of the convex combination $\boldsymbol{x}_T$:

$$\hat{\boldsymbol{x}} = (1 - p)\,\boldsymbol{x} + p\,\boldsymbol{x}_T \quad \text{with} \quad p \sim \text{Beta}(\alpha, \beta), \quad (2)$$

where the parameters $\alpha$ and $\beta$ control the relative importance of the transformed vs. original image (when $\alpha > \beta$, the values of the pixels of the transformed $\boldsymbol{x}_T$ are given more importance than the values of the pixels of the original image $\boldsymbol{x}$).

In the next section, we identify the parameter values for the dataset-specific shifts that arise for each validation set of C-CCM [13].

### III. EXPERIMENTAL EVALUATION

We conduct experiments on the C-CCM dataset [13] using a ResNet-18 [24]. From the C-CCM pre-trained models provided by [13], we evaluate the ones trained with ST (baseline), ST→FT, and AT→FT, with the latter currently being the best one for classifying C-CCM. Furthermore, we train a model directly on C-CCM using our Principal Augmentation (PA) scheme and, finally, we also explore the combination of fine-tuning an adversarially trained model [25] with PA. We denote this strategy as AT→PA. We evaluate and compare the different methods on the different splits of C-CCM. All model definitions and training procedures are implemented in PyTorch [26].

### A. Settings and analysis of the effect of the parameter values

**Split-specific parameters** The shifts of splits $S_1$ and $S_2$ are mostly related to the shape of the containers. Hence, we

---

[1]Note, that, C-CCM contains crops of images around the object of interest. For explicitly transforming only the container in a real human-to-robot application, one might need to first segment the region of interest with the container.
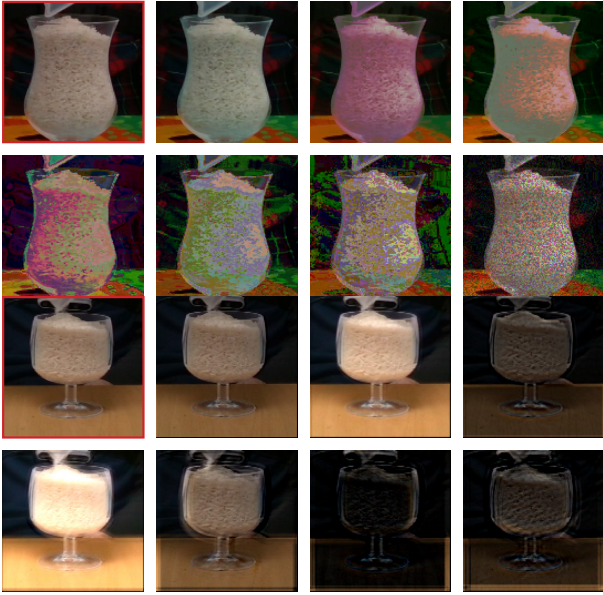
Fig. 2. Sample transformed images using color jittering (top two rows) and spectral filtering (bottom two rows) with varying smoothness level $K_\gamma$ and $K_\omega \times K_\omega$, respectively. First row, from left to right: original image, transformed image with $K_\gamma = 2, 5, 10$; second row: $K_\gamma = 20, 40, 100, 300$; third row: original image, transformed image with $K_\omega = 3, 5, 7$; fourth row: $K_\omega = 9, 11, 13, 15$.

TABLE I
VALIDATION ACCURACY OF A RESNET-18 ON THE C-CCM DATASET
SPLITS ($S_1, S_2, S_3$), WHEN THE COMPOSITION DEPTH $m$ INCREASES.
HERE, THE TRANSFORMATION WIDTH IS FIXED TO $n = 3$.

| $m$ | $S_1$ | $S_2$ | $S_3$ |
|---|---|---|---|
| 1 | 82.69 | **73.42** | 67.91 |
| 2 | 83.16 | 70.95 | 65.90 |
| 3 | **84.93** | 68.92 | **75.03** |

would like to enforce smooth, yet strong, diffeomorphisms that are able to alter the shape of the whole container so it becomes as narrow as a champagne flute, or just a part of it so it resembles the stem of a cocktail glass (see Fig. 1). In fact, for a fixed value of smoothness $K_\tau$ the authors in [27] propose to randomly sample the strength $\sigma_\tau^2$ from a specific interval, such that the resulting diffeomorphism remains bijective. In practice, for smaller values of $K_\tau$ (smoother), larger values of $\sigma_\tau^2$ are allowed to be sampled. Hence, we decided to set $K_\tau = 10$ and let $\sigma_\tau^2$ to be properly sampled during training. In practice we observed that $K\tau \in [10, 20]$ still leads to good results. The shifts of split $S_3$ are mostly related to the color and frequency content of the containers (i.e., red and green glass). We therefore focus on the color and spectral transforms (see Fig. 2). For the smoothness parameter $K$ of these transformations, we keep the original values [19]: $K_\gamma = 500$ for the color domain and $K_\omega = 3$ for the spectral domain. As for the parameter strength, since very strong changes could remove task-specific information in the images, we only slightly manipulate the color of the pixels and the frequency information of the images, and hence we set $\sigma_\gamma^2 = 0.001$ and $\sigma_\omega^2 = 0.01$, respectively.
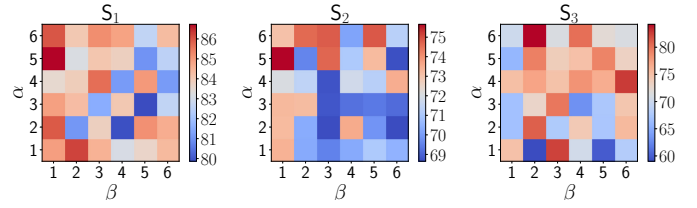


Fig. 3. Effect of the $\mathrm{Beta}(\alpha, \beta)$ distribution on the validation accuracy of a ResNet-18, trained with our principle augmentation scheme, on each split of C-CCM. Note that $\alpha > \beta$ imposes more importance to the pixels of the transformed image, while $\alpha < \beta$ to the pixels of the original image.

**Mixing parameters** It is reasonable to assume that in Eq. (1) we must set $n > 1$ in order to increase the diversity of the generated transformed instances. We use the width of the original implementation ($n = 3$). In general, it is not always possible to determine the exact outcome of the composition determined by the value of $m$, and its impact on the overall performance. Hence, we let the mixing coefficient $p$ of Eq. (2) to be uniformly sampled ($\alpha = \beta = 1$) and perform a sensitivity analysis on the values of $m$. The performance of a ResNet-18 [24] on each dataset split is shown in Tab. I: while for $S_1$ and $S_3$, increasing the depth significantly improves the performance, the opposite happens for $S_2$, suggesting that that applying multiple transformations on the image degrades some important information. Then, for the best values of $m$ in Tab. I, we explore the effect of the mixing coefficient $p$ of equation Eq. (2). We focus on the parameters of the $\mathrm{Beta}$ distribution, which control the relative importance of the pixels of $\boldsymbol{x}$ or $\boldsymbol{x}_T$. Since the classifier overfits to the training data we would expect that more importance on $\boldsymbol{x}_T$ might be necessary. To that end, we perform a sensitivity analysis on the values of $\alpha$ and $\beta$, by measuring the performance of the network on their different combinations. The results are shown in Fig. 3: on every dataset split, the highest validation accuracy is achieved when more importance is given to the pixels of the transformed image. In particular, on $S_1$ and $S_2$, the best performance (86.73% and 75.66% respectively) is achieved for $\mathrm{Beta}(5, 1)$, while on $S_3$ (84.21%) it is achieved for $\mathrm{Beta}(6, 2)$.

**Training and validation** For PA and AT→PA strategies we train or fine-tune the classifier for 50 epochs, using a cross-entropy loss [28] and stochastic gradient descent. The maximum learning rate for updating the weights is set to 0.05 and 0.005 when performing PA and AT→PA respectively. The learning rate decays linearly during training. Note that the models we evaluate are the ones that achieve the highest validation accuracy (early-stopping). For dealing with class imbalances, the training images in a batch are randomly sampled with probabilities that are inversely proportional to the number of images of each class. For the case of AT→PA, since there are multiple source models adversarially trained with perturbations of different strength $\epsilon$, we decided to choose those that lead to the highest validation accuracy. Hence, for $S_1$ we select a network trained with $\epsilon = 0.05$, while for $S_2$ and $S_3$ a network trained with $\epsilon = 0.5$. Recall that, for the AT→FT models used in [13] the selected values of $\epsilon$ were 0.05, 1 and 0.5 for each dataset split respectively.
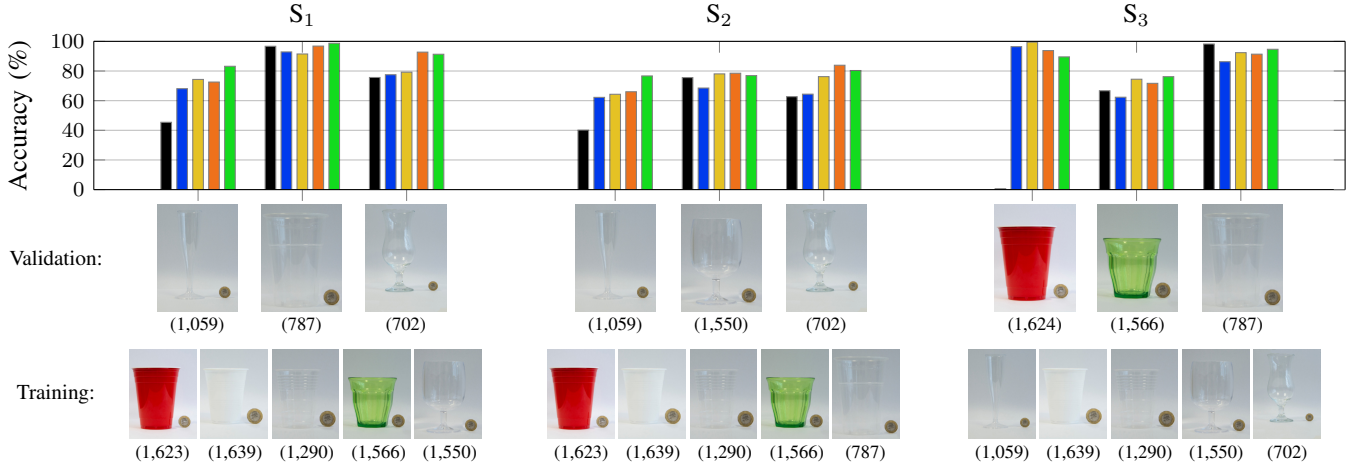
Fig. 4. Per-container filling level classification accuracy (top) on the three different dataset splits (bottom) of the C-CCM dataset. Parentheses denote the number of images for each type of container. Legend: ▬ ST$^\dagger$, ▬ ST→FT$^\dagger$, ▬ AT→FT$^\dagger$, ▬ PA, ▬ AT→PA. $^\dagger$Values taken from [13].

## B. Classification results and discussion

Fig. 4 shows the classification performance of different strategies on the three configurations, $S_1$, $S_2$ and $S_3$. The results indicate that pre-training might not be necessary, since properly tuning data augmentation to compensate for dataset-specific distribution shifts can improve performance, while requiring a lower computational cost than using transfer learning. PA requires only $1.2\times$ additional training time compared to ST on C-CCM, which is many orders of magnitude lower than (adversarially) training a model on ImageNet for transfer learning. When training time is not an issue, transfer learning with AT at the source domain combined with PA (AT→PA) generally improves performance. Note that when the performance of ST is low, all strategies lead to significant improvements; whereas when ST performs well, AT→FT has an insignificant contribution or decreases the final performance.

For $S_1$, the low performance of ST on the champagne flute (left) is improved by both AT→FT and PA, and even more so by AT→PA, suggesting that diffeomorphisms compensate for the unique narrow shape of the flute. The accuracy of ST on the beer cup (middle) is high, due to the shape similarity of the data in the training set (e.g. small transparent cup). AT→FT causes a small accuracy drop, whereas PA retains the performance and AT→PA improves it. The accuracy of ST on the cocktail glass (right) is slightly improved with AT→FT and considerably improved by PA and AT→PA. Although there is another container with a stem in the training set (wine glass), it seems that the introduced diffeomorphisms better compensate for the different shape above the stem of the cocktail glass.

For $S_2$, the accuracy of all strategies on the champagne flute (left) and the cocktail glass (right) is somehow similar in trend to that on $S_1$. Note that there are no containers with a stem in the training set. Yet, the performance on the wine glass (middle) is similar for most strategies, which might be due to the similarity of its shape above the stem with the other

transparent cups in the training set.

For $S_3$, there is no colored container in the training set. ST is unable to generalize for the red cup (left), unlike AT→FT, PA and AT→PA. Still, the accuracy with data augmentation is not on the same level as with AT→FT, which sets this specific container case as an example of the benefits of adversarially pre-training the network on a large and diverse source dataset. As for the green glass (middle) AT→FT increases on ST, similarly to AT→PA. Finally, the accuracy of ST on the beer cup (right) is high and the other strategies cannot reach that level, with AT→PA featuring the lowest performance drop.

## IV. CONCLUSION

We investigated the impact of data augmentation on classifying the filling level of a container, when the containers in the test-time images have different properties than those in the training set. We compared transfer learning – with or without adversarial training [13] – and a principled augmentation approach that explicitly operates on the geometry, color and spatial frequencies of the training images [19] in order to generalize the shape, color, and spectral content of an available, limited training dataset. We showed that the principled augmentation can either replace transfer learning approaches, which are computationally more expensive, or be combined with adversarial transfer learning to improve its performance.

As future work, we will incorporate new transformations to compensate for other types of distribution shifts related to transparencies and occlusions. Furthermore, we will extend our analysis to other datasets and settings.

REFERENCES

[1] C. Tan, F. Sun, T. Kong, W. Zhang, C. Yang, and C. Liu, "A survey on deep transfer learning," in *Int. Conf. on Artificial Neural Networks*, 2018, pp. 270–279.

[2] M. Huh, P. Agrawal, and A. A. Efros, "What makes ImageNet good for transfer learning?" in *Neural Inf. Process. Syst. Workshop on Large Scale Computer Vision Systems*, Dec. 2016.

[3] S. Kornblith, J. Shlens, and Q. V. Le, "Do better ImageNet models transfer better?" in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2019.

[4] J. Xue, H. Zhang, and K. Dana, "Deep texture manifold for ground terrain recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018.

[5] A. Farhadi, I. Endres, D. Hoiem, and D. Forsyth, "Describing objects by their attributes," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2009.

[6] S. S. Sajjan, M. Moore, M. Pan, G. Nagaraja, J. Lee, A. Zeng, and S. Song, "ClearGrasp: 3D shape estimation of transparent objects for manipulation," in *Proc. IEEE Int. Conf. Robotics Autom.*, Jun. 2020.

[7] A. Vedaldi, S. Mahendran, S. Tsogkas, S. Maji, R. Girshick, J. Kannala, E. Rahtu, I. Kokkinos, M. B. Blaschko, D. Weiss, B. Taskar, K. Simonyan, N. Saphra, and S. Mohamed, "Understanding objects in detail with fine-grained attributes," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2014.

[8] C. Shorten and T. M. Khoshgoftaar, "A survey on image data augmentation for deep learning," *Journal of Big Data*, vol. 6, no. 1, 2019.

[9] D. Hendrycks, N. Mu, E. D. Cubuk, B. Zoph, J. Gilmer, and B. Lakshminarayanan, "AugMix: A simple method to improve robustness and uncertainty under data shift," in *Proc. Int. Conf. Learning Represent.*, Apr. 2020.

[10] R. Sanchez-Matilla, K. Chatzilygeroudis, A. Modas, N. Ferreira Duarte, A. Xompero, P. Frossard, A. Billard, and A. Cavallaro, "Benchmark for human-to-robot handovers of unseen containers with unknown filling," *IEEE Robotics Autom. Lett.*, vol. 5, no. 2, Apr. 2020.

[11] A. Xompero, R. Sanchez-Matilla, A. Modas, P. Frossard, and A. Cavallaro, "Multi-view shape estimation of transparent containers," in *Proc. IEEE Int. Conf. Acoustics, Speech Signal Process.*, May 2020.

[12] R. Mottaghi, C. Schenck, D. Fox, and A. Farhadi, "See the glass half full: Reasoning about liquid containers, their volume and content," in *Proc. IEEE Int. Conf. Comput. Vis.*, Oct 2017.

[13] A. Modas, A. Xompero, R. Sanchez-Matilla, P. Frossard, and A. Cavallaro, "Improving filling level classification with adversarial training," in *Proc. IEEE Int. Conf. Image Process.*, 2021, pp. 829–833.

[14] C. Schenck and D. Fox, "Visual closed-loop control for pouring liquids," in *Proc. IEEE Int. Conf. Robotics Autom.*, May 2017.

[15] C. Do, T. Schubert, and W. Burgard, "A probabilistic approach to liquid level detection in cups using an RGB-D camera," in *Proc. IEEE Int. Conf. Intell. Robot Syst.*, Oct. 2016.

[16] C. Do and W. Burgard, "Accurate pouring with an autonomous robot using an RGB-D camera," in *Proc. Int. Conf. Intell. Auton. Syst.*, Jul. 2018.

[17] J. Deng, W. Dong, R. Socher, L.-J. Li, L. Li, and L. Fei-Fei, "ImageNet: A large-scale hierarchical image database," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2009.

[18] D. Hendrycks and T. Dietterich, "Benchmarking neural network robustness to common corruptions and perturbations," in *Proc. Int. Conf. Learning Represent.*, May 2019.

[19] A. Modas, R. Rade, G. Ortiz-Jiménez, S.-M. Moosavi-Dezfooli, and P. Frossard, "PRIME: A few primitives can boost robustness to common corruptions," *arXiv:2112.13547*, Dec. 2021.

[20] A. Xompero, R. Sanchez-Matilla, R. Mazzon, and A. Cavallaro, "CORSMAL Containers Manipulation," 2020, (1.0) [Dataset]. Queen Mary University of London. https://doi.org/10.17636/101CORSMAL1. [Online]. Available: http://corsmal.eecs.qmul.ac.uk/containers_manip.html

[21] A. Madry, A. Makelov, L. Schmidt, D. Tsipras, and A. Vladu, "Towards deep learning models resistant to adversarial attacks," in *Proc. Int. Conf. Learning Represent.*, Apr. 2018.

[22] C. Szegedy, W. Zaremba, I. Sutskever, J. Bruna, D. Erhan, I. Goodfellow, and R. Fergus, "Intriguing properties of neural networks," in *Proc. Int. Conf. Learning Represent.*, Apr. 2014.

[23] S. Moosavi-Dezfooli, A. Fawzi, and P. Frossard, "DeepFool: A simple and accurate method to fool deep neural networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2016.

[24] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2016.

[25] H. Salman, A. Ilyas, L. Engstrom, A. Kapoor, and A. Madry, "Do adversarially robust ImageNet models transfer better?" in *Adv. Neural Inf. Process. Syst.*, Dec. 2020.

[26] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, A. Desmaison, A. Kopf, E. Yang, Z. DeVito, M. Raison, A. Tejani, S. Chilamkurthy, B. Steiner, L. Fang, J. Bai, and S. Chintala, "PyTorch: An imperative style, high-performance deep learning library," in *Adv. Neural Inf. Process. Syst.*, Dec. 2019.

[27] L. Petrini, A. Favero, M. Geiger, and M. Wyart, "Relative stability toward diffeomorphisms indicates performance in deep nets," in *Adv. Neural Inf. Process. Syst.*, Dec. 2021.

[28] S. Shalev-Shwartz and S. Ben-David, *Understanding Machine Learning: From Theory to Algorithms*. Cambridge University Press, 2014.