Deep Learning for Autonomous Cars
Semi-Supervised Learning in Simulation

Hye Jin Kim

January 6, 2022

Abstract

Semi-Supervised Learning is a highly efficient method to leverage unlabeled data in cases where labeling data is expensive for large datasets. This paper aims to investigate whether Semi-Supervised Learning algorithms, namely MixMatch and FixMatch from Google Research, can be used to automate a car agent in OpenAI's car-racing simulation environment. The performance of various models is tested through statistical analysis for 50 consecutive trials in the simulation. Convolutional Neural Networks trained using FixMatch with Selective MixUp, a novel modification of the original FixMatch algorithm, generally perform better than models trained using the MixMatch approach. FixMatch with Selective MixUp using 800 labels (10% of the total training data) achieved an average car-racing score of 719 and a median of 775.71 for 50 successive trials in the simulation. This result is comparable to the performance of Supervised Learning with 8000 labels.

1 Introduction

Ever since a Deep Convolutional Neural Network (CNN) won the ImageNet image classification challenge [2], this type of Artificial Neural Network (ANN) is commonly used for computer vision applications because of its scalability resulting in state-of-the-art performance. There are various practical uses of CNN image classification ranging from static tasks such as labeling medical images for diagnosing patients to dynamic tasks such as identifying environment orientations for controlling the movement of self-driving cars. The current paper demonstrates how CNNs can be used for autonomous driving in a simple 2D car-racing simulation environment from OpenAI: Car-Racing-v0.

There are several challenges to overcome for successful autonomous driving with CNNs. In real life, trained models for self-driving cars need high accuracy; otherwise, the cost is devastating. To increase accuracy, models can train on larger datasets. However, this comes with the resource expense of labeling a substantial amount of images. SSL methods are beneficial in such scenarios because they leverage unlabeled data by adding a loss term calculated on the unlabeled data, alleviating the need for labeling such large datasets. This paper implements two well-known SSL algorithms from Google Research—MixMatch and FixMatch—in the Car-Racing-v0 environment.

The current paper aims to compare the SSL methods of MixMatch and FixMatch to the baseline Fully-Supervised Behavioral Cloning by testing the CNN models' autonomous driving performance in the OpenAI simulation. These findings can give implications for applying SSL methods to automate cars and other dynamic computer vision tasks.

2 Theory

2.1 Supervised Learning: Behavioral Cloning

The Behavioral Cloning algorithm is an imitation learning algorithm, ideal for creating a baseline CNN car-driving model because expert demonstrations from human players are easily obtainable. This approach is much more straightforward than manually specifying a reward function for the model to learn. Expert demonstrations are collected by capturing image frames of the OpenAI environment as a human expert plays the car-racing game. These demonstrations are treated as i.i.d. (independent and identically distributed) state-action pairs [3]. In other words, each image frame (state) is paired with a certain car behavior (action). The CNN model learns the expert's policy by minimizing the loss calculated from the difference between the model's action prediction and the ground truth when fed states of the environment.

2.2 SSL: MixMatch and FixMatch

The MixMatch method utilizes three different approaches for estimating the additional loss term for unlabeled data: consistency regularization, entropy minimization, and generic regularization [1]. Consistency regularization encourages the model to maintain the same output distributions when augmentation is applied to the inputs. Entropy minimization increases the model's confidence in unlabeled data output predictions. Generic regularization helps avoid overfitting the training data. Compared to MixMatch, FixMatch is a simple yet more sophisticated SSL method that only combines two approaches: pseudo-labeling, which implicitly minimizes entropy, and consistency regularization [4].

3 Methods

3.1 Dataset

Expert demonstrations are collected by human players controlling the car agent using the Up (Accelerate), Down (Brake), Left, and Right keyboard arrow keys in the Car-Racing-v0 simulation. Only the demonstrations from trials with total rewards above 850 are considered to secure reliable image frames for a high-performing CNN model. For a simple expert's policy, only the image frames paired with a single action of the car agent are used (i.e., only four action classes—Accelerate, Brake, Left, and Right—are considered, and combinations such as 'Left and Accelerate' and no-action states are excluded). Furthermore, the first 30 image frames of each trial are removed for consistency, and the class distribution is balanced for sufficient learning of each class. The final dataset used consists of 10000 image frames with a class distribution of 37% Acceleration, 8% Brake, 28% Left, and 27% Right.

3.2 Data Preprocessing

For all learning methods, the same data preprocessing is applied. All images are normalized and grayscaled to minimize feature bias and reduce training time. They are also downscaled from an image size of 600 x 400 to 96 x 96 for better compatibility with the Car-Racing-v0 environment. Using a train/validation/test ratio of 8:1:1 respectively, the training dataset is preprocessed further by masking the bottom bar of the image. This step prevents the model from learning supplementary indicators.

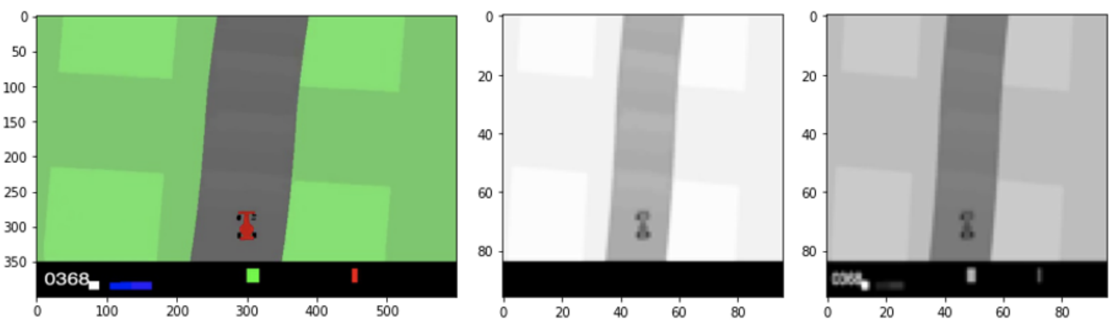

Figure 1: Example outputs after data preprocessing. From left to right: raw image, image after data preprocessing for the training dataset, and image after data preprocessing for the validation and test dataset.

To compare the performance of the SSL methods to Behavioral Cloning (with which 100% of the training data are labeled), the ratio between labeled and unlabeled images during model training is adjusted. Starting from a 50:50 ratio, the proportion of labeled images is reduced by 10% until only 10% of the training data are labeled. (i.e., the five labeled-to-unlabeled image distributions used are 50:50, 40:60, 30:70, 20:80, and 10:90).

3.3 CNN Model

The overall architecture design of the CNN model is simple to reduce computational costs. The CNN has two convolutional layers, both with a kernel size of 3 x 3 pixels and 32 output channels; each layer is followed by a max-pooling layer. Batch normalization is applied to the input layer and the first of two fully-connected linear layers. A dropout layer with a dropout rate of 0.5 is added between the two linear layers for regularization.

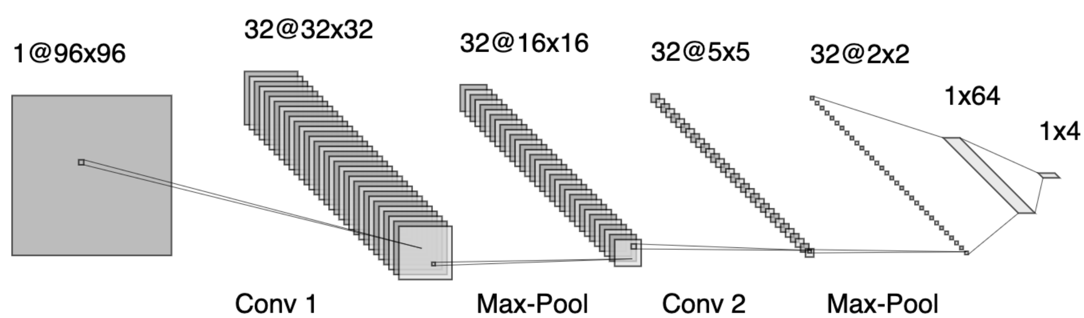

Figure 2: The CNN architecture. The main structure is the two convolutional blocks in which each convolutional layer is followed by a max-pooling layer. The two vectors at the end represent the respective output shape of each fully-connected layer.

For all learning methods, the CNN model is trained using the Adam optimizer with a learning rate of 0.0001. In the case of Supervised Learning with Behavioral Cloning, cross-entropy loss is used for the loss function criterion.

3.4 SSL: Utilities and Loss Function

3.4.1 MixMatch

Several utilities are used in the MixMatch algorithm to calculate the unlabeled loss: 1) data augmentation for consistency regularization 2) a combination of label guessing and sharpening for entropy minimization and 3) MixUp for general regularization.

K augmentations are randomly selected and applied to both labeled and unlabeled data from a pool of augmentations. The augmentations adjust the brightness, contrast, saturation, rotation, or gamma level of the images. Because the car-racing dataset used is direction-sensitive, precaution is taken to ensure that the labels are maintained after the augmentations. In other words, image augmentations such as horizontal flip, which alters a Left-labeled image into a Right-labeled image, are not used. Label guesses for unlabeled data are generated by taking the average of the K augmented unlabeled data predictions and minimizing the entropy of the distribution with a sharpening function. Finally, MixUp mixes the augmented labeled examples and unlabeled examples with label guesses through a process of convex combination, producing new examples. The total loss is calculated from the summation of the cross-entropy loss on the mixed labeled data and the squared L2 loss on the mixed unlabeled data.

An ablation study of MixMatch without the MixUp utility is performed to investigate the importance of the MixUp utility. Furthermore, a modified version of MixUp—Selective MixUp—is proposed to tailor the direction-sensitive characteristic of the dataset. In Selective MixUp, only examples with the same class label (or guessed label in the case of unlabeled data) are mixed. Similar to the augmentation choices, this step encourages a clear separation of the class labels.

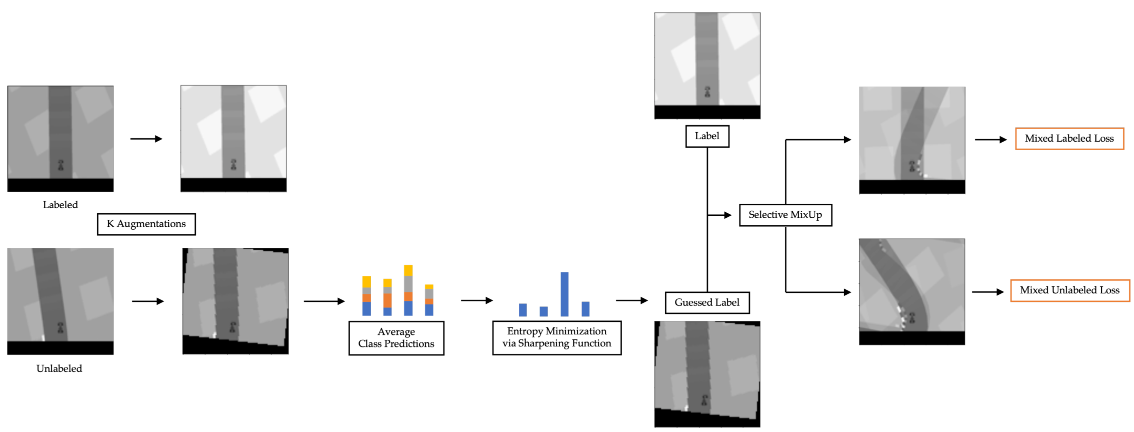

Figure 3: Diagram of MixMatch with Selective MixUp. K augmentations are applied to labeled and unlabeled data. Average class predictions are made for augmented unlabeled data. Guessed labels are then produced by minimizing the distribution entropy via a sharpening function. Finally, Selective MixUp is applied to augmented labeled data and augmented unlabeled data with guessed labels.

3.4.2 FixMatch

In FixMatch, two utilities are used to calculate the unlabeled loss: 1) data augmentation for consistency regularization and 2) pseudo-labeling for entropy minimization. Pseudo-labels are obtained from the model's prediction when fed weakly-augmented unlabeled inputs. High confidence pseudo-labels are used as targets for a strongly-augmented version of the same inputs. Similar to

MixMatch, augmentation selection is done carefully to avoid altering the integrity of the labels. The total loss is calculated from the summation of the cross-entropy loss on the weakly-augmented labeled data and the cross-entropy loss on the strongly-augmented unlabeled data.

In the current paper, a novel variation of the FixMatch algorithm, which utilizes the entropy minimization method of MixMatch along with Selective MixUp, is introduced. The labeled cost is the same, but the unlabeled loss is calculated on the strongly-augmented mixed unlabeled data using cross-entropy.

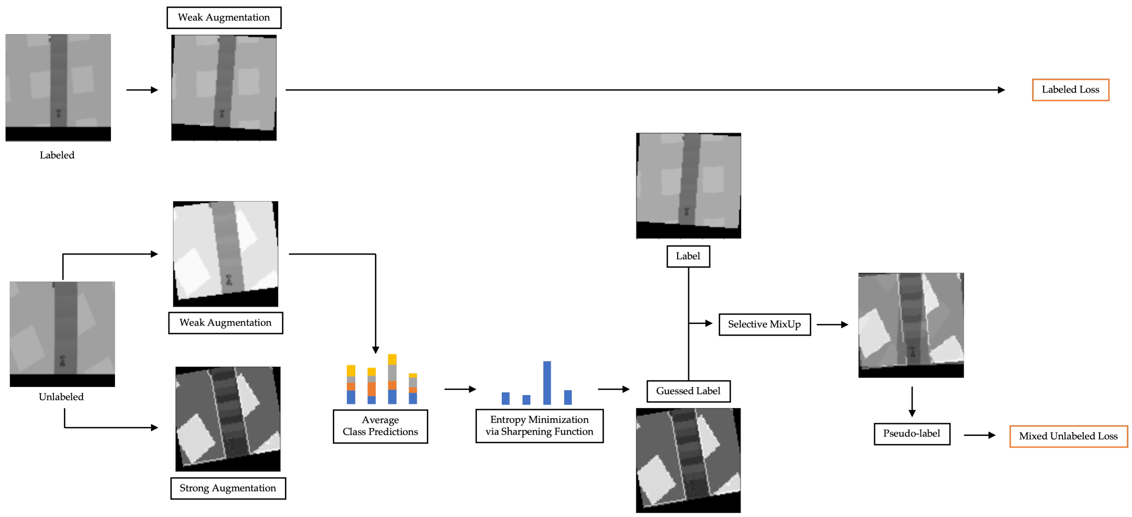

Figure 4: Diagram of FixMatch with Selective MixUp. Weakly-augmented labeled data is used to calculate the labeled loss. Average class predictions are made for weakly-augmented unlabeled data. Then, guessed labels are generated by using a sharpening function to minimize the entropy. Selective MixUp is applied to weakly-augmented labeled data and strongly-augmented unlabeled data with guessed labels. Finally, pseudo-labels are produced from the strongly-augmented mixed unlabeled data.

4 Results

4.1 Brute Force and Stop

For each learning method, the CNN model is trained with a batch size of 32 for 50 epochs. First, the Fully-Supervised Behavioral Cloning baseline model trained with 8000 labeled data is tested in the Car-Racing-v0 environment. The goal is to secure a good baseline model with a high average score of above 700 to ensure that the model will also train well with SSL methods.

In the initial simulation runs, although the baseline model could classify the image frames of the car-racing environment into each of the four class labels accordingly, a problem occurred where the car agent sometimes stopped at turns and did not accelerate any further. Therefore, a Brute Force code is introduced in which the car is accelerated for five image frames if there is more than one consecutive class prediction of Brake. This code prevents the car from stopping completely. Similarly, a Brute Stop code is also used to brake the car for five image frames if the last 50 class predictions are Acceleration. This code helps to reduce the occasional aggressive driving behavior of the car agent.

4.2 Simulation Statistics

To test the performance of the various trained CNN models in the simulation, each model runs through 50 successive trials. Statistical analysis of the trials is conducted by estimating and comparing the scores' mean, median, and interquartile range. Scores below 0 are considered incomplete trials and thus, excluded from the calculations. Furthermore, only the models that completed 50 consecutive runs with less than 15 incomplete trials are evaluated. These models are the ones trained using the SSL methods of either MixMatch without MixUp (MM), MixMatch with Selective MixUp (MMSM), or FixMatch with Selective MixUp (FMSM).

Table 1: Simulation statistics for 50 successive trials, mainly comparing the successful SSL methods. The distribution indicates the ratio of labeled-to-unlabeled data, and the labeled data signifies the number of labeled data used for training. The MixMatch without MixUp results for the 10:90 distribution is excluded because the model was unable to successfully finish 50 consecutive trials.

Distribution (Labeled Data)	Methods	Mean	Median	Interquartile Range
100:0 (8000)	SL	742.72	767.26	71.90
50:50 (4000)	MM	765.07	783.19	63.93
	MMSM	639.29	645.49	216.81
	FMSM	783.77	810.49	99.05
40:60 (3200)	MM	725.30	750.68	151.21
	MMSM	758.97	781.36	88.63
	FMSM	666.66	697.31	182.01
30:70 (2400)	MM	711.71	739.26	71.86
	MMSM	574.73	611.71	183.39
	FMSM	740.31	754.14	116.64
20:80 (1600)	MM	588.01	639.33	198.51
	MMSM	711.06	715.93	52.95
	FMSM	677.68	723.03	133.19
10:90 (800)	MM	-	-	-
	MMSM	595.57	606.54	180.60
	FMSM	719.00	775.71	95.89

Among the SSL methods, FixMatch with Selective MixUp is in general, the best SSL method for automating the car agent in the Car-Racing-v0 environment as it consistently has a high maximum, mean, and median score for all of the labeled-to-unlabeled distributions. Even in the 10:90 distribution condition (800 labeled data), FixMatch with Selective MixUp has an average score of 719 and a median of 775.71. However, when the number of labeled data is reduced, the stability of the SSL models decreased; the number of outliers and the interquartile range increased.

MixMatch with Selective MixUp also generally performed well. It is even better than FixMatch with Selective MixUp in the 40:60 distribution condition. However, this may be due to the random sampling of unlabeled data during training. The performance of MixMatch without MixUp decreases significantly as the number of labeled data is reduced, and is unable to successfully finish 50 consecutive trials in the 10:90 condition. The results indicate that Selective MixUp is somehow crucial in the performance of the CNN model trained with SSL methods in the Car-Racing-v0 environment. Selective MixUp not only increases the generalizability of the model but also encourages the distinct separation of the class labels during training.

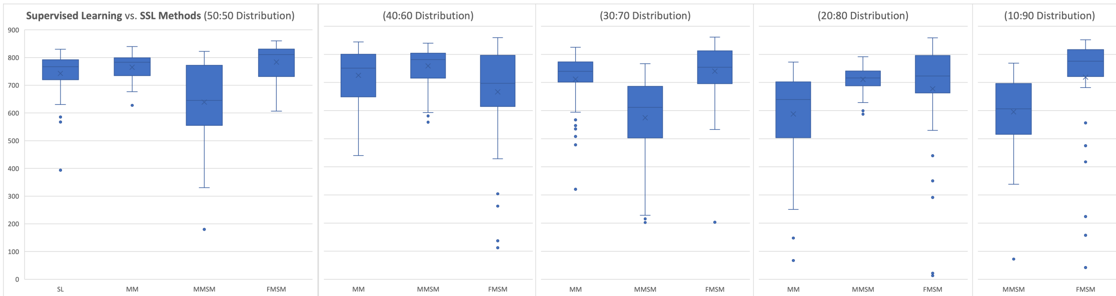

Figure 5: Boxplot displaying the various trained CNN models' simulation statistics for 50 consecutive trials. Supervised Learning indicates the CNN model trained using the Behavioral Cloning method with 8000 labeled data. The distribution ratios for the SSL methods signify the labeled-to-unlabeled image distributions.

4.3 Supervised Learning with Reduced Dataset Size

To further compare the performance between Supervised Learning and SSL, Supervised Learning models trained with reduced dataset sizes are tested in the Car-Racing-v0 environment. The size of the dataset is reduced to match the number of labeled images in the various labeled-to-unlabeled distributions. Although the dataset size is modified, the train/validation/test ratio of 8:1:1 is maintained. For example, if the dataset is reduced by 50%, the number of training images is decreased to 4000, and the number of validation and test images is each decreased to 500. Based on the simulation statistics, the best SSL method FixMatch with Selective MixUp is used to compare the results of the Supervised Learning models trained with reduced dataset sizes.

Table 2: Simulation statistics for 50 successive trials, mainly comparing Supervised Learning to the best performing SSL method FixMatch with Selective MixUp. The same number of labeled data is used for each comparison.

Labeled Data	Methods	Mean	Median	Interquartile Range
8000	SL	742.72	767.26	71.90
4000	SL	747.63	786.56	120.29
	FMSM	783.77	810.49	99.05
3200	SL	729.21	750.69	98.22
	FMSM	666.66	697.31	182.01
2400	SL	751.81	777.05	103.04
	FMSM	740.31	754.14	116.64
1600	SL	731.86	750.49	121.78
	FMSM	677.68	723.03	133.19
800	SL	737.85	780.36	87.60
	FMSM	719.00	775.71	95.89

The performance of Supervised Learning with reduced dataset size is high, even with 800 labeled data. Overall, the results are similar to FixMatch with Selective MixUp which may lead to the conclusion that the SSL methods are only benefiting from the labeled data. However, it is important not to draw such a quick conclusion as there are several factors to consider. The main purpose of SSL methods is not to be better than Supervised Learning, but to be comparable, mitigating the

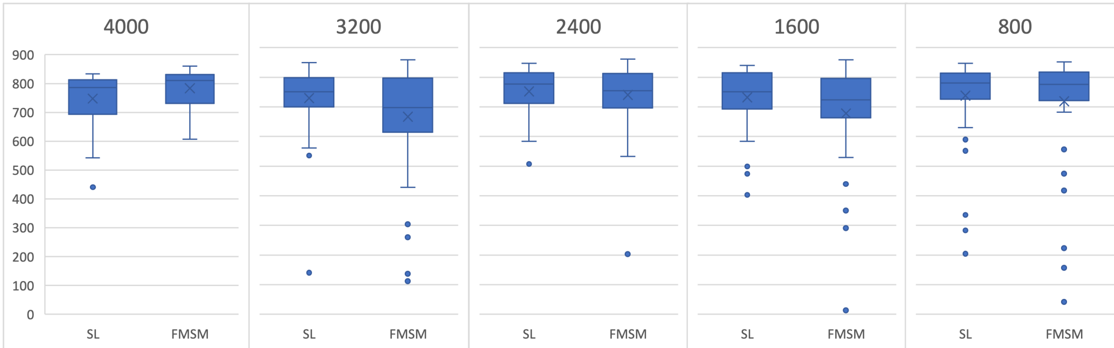

Figure 6: Boxplot displaying the Supervised Learning and SSL method of FixMatch with Selective MixUp simulation statistics for 50 consecutive trials. The number above the boxplots (i.e., 4000, 3200, 2400, 1600, and 800) represents the number of labeled data used for training.

cost of labeling data in cases where the dataset size is very large (e.g., more than 50000 images). The current literature on SSL techniques generally use more complex models to train on larger datasets. Therefore, it is probable that using a more complex CNN model with a larger dataset size is better for comparing Supervised Learning and SSL methods.

Furthermore, SSL methods may be more applicable for datasets with classes that are better distinguishable. For example, the appropriate label for a road image slightly oriented in the left direction may be Accelerate or Left. This is a more challenging classification task compared to differentiating between a cat and a dog. Nonetheless, the fact that the SSL methods are comparable to Supervised Learning while guessing labels for 4000 or more unlabeled data is promising.

5 Conclusion and Future Studies

In this report, the Semi-Supervised Learning algorithms MixMatch and FixMatch are implemented in the OpenAI Car-Racing-v0 environment. To make the SSL methods more suitable for the simulation environment, novel variations of the two SSL methods using the Selective MixUp utility are introduced. SSL algorithms that implemented the Selective MixUp utility generally scored higher in the Car-Racing-v0 environment. This indicates that Selective MixUp may be a key component in SSL algorithms for direction-sensitive classification tasks. Among the SSL methods successfully tested in the simulation, FixMatch with Selective MixUp produced the highest average and median score using the lowest amount of labeled data. These results are comparable to the Fully-Supervised Learning method of Behavioral Cloning.

Although the results of SSL methods are comparable to Supervised Learning, the CNN models trained using SSL became more unstable as the amount of labeled data decreased. For safe driving in the real-life environment, the stability of the model's performance is important. Therefore, future work should aim to lower the model's fluctuating performance even with a small number of labeled data. For this, the complexity of the CNN and the dataset size may need to be increased, and SSL techniques such as Selective MixUp should be used and further improved for better suitability in the direction-sensitive car-racing environment.

References

[1] D. Berthelot, N. Carlini, I. Goodfellow, N. Papernot, A. Oliver, and C. A. Raffel, "Mixmatch: A holistic approach to semi-supervised learning," *Advances in Neural Information Processing Systems*, vol. 32, 2019.

[2] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," *Advances in neural information processing systems*, vol. 25, 2012.

[3] Z. Lőrincz, "A brief overview of imitation learning," Sep 2019. [Online]. Available: https://smartlabai.medium.com/a-brief-overview-of-imitation-learning-8a8a75c44a9c

[4] K. Sohn, D. Berthelot, N. Carlini, Z. Zhang, H. Zhang, C. A. Raffel, E. D. Cubuk, A. Kurakin, and C.-L. Li, "Fixmatch: Simplifying semi-supervised learning with consistency and confidence," *Advances in Neural Information Processing Systems*, vol. 33, pp. 596–608, 2020.