Learn2Augment: Learning to Composite Videos for Data Augmentation in Action Recognition Supplementary

Shreyank N Gowda[1], Marcus Rohrbach[2], Frank Keller[1], and Laura Sevilla-Lara[1]

[1] University of Edinburgh
[2] Meta AI

Summary and Outline

Section 1 explains quantitatively why we it is better to augment using pairs of videos from classes that are semantic neighbors instead of using pairs from the same class.
Section 2 explains the distillation loss mentioned in the paper.
Section 3 shows the comparison of number of augmented samples vs accuracy for different semi-supervised settings of UCF101.
Section 4 looks at some alternative approaches to the design of the selector network.
Section 5 looks at why we need to re-train the classifier network instead of jointly training the classifier and selector.
Section 6 shows some examples of selected and discarded mixed videos. Note that the discarded examples are not generated by the model at inference time and we only show it for visualization purposes.
Section 8 lists the neighboring classes used for our augmentation strategy in all settings for UCF101, HMDB51 and Kinetics100 using sen2vec [5].
Section ?? shows a comparison of the percentage of data used for validation and the change in performance of the overall model.

1 Why Not Intra-class Augmentation?

One other possibility we explored is intra-class augmentation instead of using semantic classes. However, when we followed the same procedure on 20% labeled data of UCF101 we obtain an accuracy of 41.4% in comparison to 58.9% when using semantically similar classes. Similarly, in Kinetics100 we obtain an accuracy of 50.1% and 54.4% using 5% and 10% labeled data respectively. That is 9.4% and 8.9% lower than the results using semantic neighbors. We believe there to be two main concerns in intra-class augmentation. The first is that Cutmix [6] has been shown to be an excellent regularization technique. This is aided by having samples that have soft labels (since they are a ratio of samples from different classes). However, using intra-class augmentation would force the labels

to be the same as the ground truth class. The second reason is that samples of a particular class are clips that were part of the same video. This is the case in both HMDB51 and UCF101 and not so in Kinetics100. If we cut the background from one sample and paste the foreground onto this, it results in an identical sample to the original foreground sample. This is because the background is the same in both cases. All we end up doing then is training the model on multiple instances of the same data which leads to overfitting and hence a poor accuracy at test time. However, since the results are much worse for Kinetics100 as well, we believe that this could be a smaller contributing factor.

2 Distillation Loss for Semi-Supervised Learning (SSL)

Given frame a from video v, to distill appearance information of objects of interest, we use the softmax predictions of a ResNet [3] image classifier. This network is pre-trained on Imagenet and not modified during training. Let the output of the ResNet be denoted as $h(a) \in \mathbb{R}^M$ where $M = 1000$ which is the number of classes in Imagenet. We randomly select a frame from all videos (labeled, unlabeled and augmented) for training. The classifier model in our architecture, produces an embedding $q(v) \in \mathbb{R}^M$ which is of the same dimensions and space of $h(a)$. We train $q(v)$ to match the output of $h(a)$ by using a soft cross-entropy loss that treats the ResNet outputs as soft labels. This loss $\mathcal{L}_d$ can be seen in Eq. 1. Our final loss function is a combination of $\mathcal{L}_d$ and $\mathcal{L}_s$ (categorical cross-entropy loss for video samples). This is done following the work in VideoSSL [4].

$$\mathcal{L}_d = - \sum_{v \epsilon (X \cup Z), a \epsilon v} h(a) \log \left(q(v) \right) \tag{1}$$

3 Analysis of Number of Augmented Samples

We see a common pattern when adding augmented samples to the different SSL settings. This basically refers to increasing the number of augmented samples in the training set. We see that the accuracy increases initially, reaches a peak performance and then starts dropping slowly as can be seen in Figure 1. This makes sense as we don't expect every mixed example to be helpful for training. In fact, this helps us to define ω_i for the selector. We can see Figure 1 for the results from 0 augmentations to 5000 for 10% and 20% labeled data on UCF101. The sweet spot for the 10% labeled data is around 1200 augmentations and for the 20% labeled data is around 2000 augmentations. Both of which are obtained using $\omega_i = 0.6$. We decide the value of ω_i based on these and results and use the same for HMDB51 and Kinetics100 for all settings. If we increase the value of ω_i we obtain fewer samples and decreasing the value of ω_i results in more number of samples for training. The value of ω_i thus determines the number of augmented samples and also their quality.

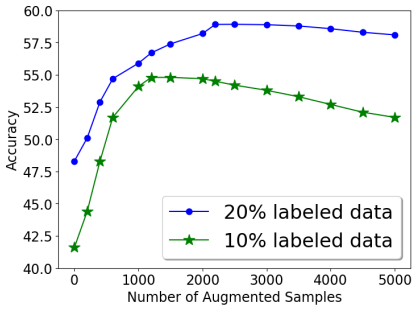

Fig. 1. Comparison of performance with increasing number of augmented samples. Results are for 10% and 20% of labeled data UCF101. We see that the performance increases initially, reaches a peak and slowly starts dropping.

4 Other Selector Choices

The design of the selector is a crucial aspect of our model. We want the selector to be able to learn what makes a good pair of videos for mixing without actually having to mix every single pair. However, for lower percentages of labeled data, we can generate all possible samples of semantic classes and convert a state-of-the art frame selection model (SMART) [1] to do sample importance instead of frame importance. We also consider a simple baseline of using a discriminator network to pick only realistic samples. We report the results in Table 1. Another approach was to randomly pick a certain amount of samples to train the classifier network.

We not only outperform all alternative approaches, we also do this by saving on both memory and computation cost. For example, in the 20 percent setting, SMART sees 99K videos and these 99k videos have to be precomputed and stored before training SMART. However, the proposed approach only needs 12K videos and outperforms SMART by up to 1.4%. This analysis is only to show a comparison to possible alternatives when storing data is feasible. The idea of trying these alternatives is only feasible in low percentage labeled data of small datasets like UCF101 and HMDB51. Even 50% labeled data in UCF101, results in having to mix over 400k videos while large scale datasets like Kinetics400 would lead to millions of mixes being needed making it practically unfeasible.

5 Why Re-train the Classifier Network?

Here, we are talking about the classifier network in our proposed architecture that the selector learns from (based on the validation loss). Training the Selector and the Classifier together is also possible. But we decide against this for 2 reasons. First, and the most important reason is that we want to save out on computational cost needed to generate an augmented sample. We showed

	50%		20%		10%		5%	
Method	Acc	SS	Acc	SS	Acc	SS	Acc	SS
Random	61.9	430K	56.2	99K	51.8	44K	42.3	9.7K
Discriminator	62.8	430K	57.3	99K	52.2	44K	41.1	9.7K
SMART [1]	68.9	430K	58.9	99K	57.8	44K	46.5	9.7K
Proposed	**72.1**	39K	**60.3**	12K	**56.1**	5.2K	**48.0**	1.2K

Table 1. Comparison of approaches for the use of Selector. All results are reported on UCF101. 'Acc' corresponds to accuracy and 'SS' corresponds to the number of mixed videos that the Selector looks at. All results are on different percentage of labeled data in UCF101.

that the selector network looks only at a fraction of samples before it understands what makes a good pair. Hence, we first train the selector by generating augmented samples taken from random samples of semantically similar classes. Once the selector is trained, we don't need to generate the mixed sample for all possible pairs and only generate the mixed samples for good pairs (the selector need not have seen these pairs before). We then augment the original dataset by samples that the selector believes will improve the classifiers performance We compare the performance of the joint training and re-training of the classifier network in Table 2. We see that re-training the classifier network always yields the best performance.

Method	50%	20%	10%	5%
Jointly trained	66.5	57.4	53.1	44.7
Retrained	**72.1**	**60.3**	**56.1**	**48.0**

Table 2. Comparison of jointly training classifier and re-training it. We see that there is a consistent large improvement in re-training the classifier.

6 Examples of Selected and Discarded Samples

To understand what made a good sample we visualize a few samples that were selected by the selector model and a few samples that were discarded. These can be seen in Figure 2. The samples are displayed as 4 frames for better visualization. Based on the small subset of examples seen, we believe that for good pairs to be selected some of the criteria could be coherent inpainting, similar camera movement, not too drastic a background change.

We see some samples of discarded examples in Figure 3. Based on the small subset of examples seen we think possible bad pairs are due to bad video compositing (example 2 in Figure 3), varying camera movements (example 3 in

Fig. 2. Visualizing selected examples. From top to bottom as (foreground, background) pairs: (flic-flac, cartwheel), (smile, laugh), (playing violin, playing cello), (front crawl, swimming backstroke). The first two are examples from HMDB51 and the last two from UCF101.

Figure 3) or a drastic change in background (example 1 in Figure 3). These are however based on the few examples we see.

Fig. 3. Visualizing discarded examples. From top to bottom as (foreground, background) pairs: (somersault, diving), (climbing stairs, falling floor), (baby crawling, walking dog), (hammering, hammer throw).

7 Effect of Semantic Match in generalization ability.

We test the generalization ability of the semantic matching by comparing it with random matching which would correspond to row 4 of Table 1 in the main paper. We observe that the performance does decrease. To strengthen this test, we tried the same experiment in the FSL setting, which is an extreme case for generalization. We augment data for two different methods, using the proposed L2A, using both semantic and random matching of classes. We observe that even in this setting, which is the most susceptible to overfitting, the semantic matching outperforms random matching. We will add this to the final version.

8 Semantic Classes

In order to better understand what are the semantic neighbors used for the video compositing process we list them out here. While many of them are bidirectional (both classes have each other as neighbors) a few of them are not.

Method	Class Matching	1-shot	3-shot	5-shot
C3D-PN	Random	28.1	42.9	47.7
C3D-PN	Semantic	29.9	44.5	50.8
TRX	Random	33.5	49.9	60.3
TRX	Semantic	35.0	51.1	62.1

Table 3. Results on FSL using the proposed Semantic Matching vs random matching using the TruZe [2] split.

This is expected as not all classes have a strong semantic neighbor. For example, in UCF101, the class "brushing teeth" has a semantic neighbor as "head massage". However, "head massage" has a bidirectional relationship with "haircut". Bidirectional pairs are written in bold. We list semantic neighbors of UCF101, HMDB51 and Kinetics100 using Sen2vec [5]. The pairs are written in the order (foreground, background). However, for bidirectional pairs the reverse applies too.

8.1 UCF101

(Apply Eye Makeup, Apply Lipstick), (Archery, Fencing), (Baby Crawling, Walking with a Dog), **(Balance Beam, Floor Gymnastics)**, **(Band Marching, Military Parade)**, **(Baseball Pitch, Golf Swing)**, **(Basketball Shooting, Basketball Dunk)**, **(Bench Press, Clean and Jerk)**, **(Billiards Shot, Table Tennis Shot)**, **(Bodyweight Squats, Lunges)**, **(Boxing Punching Bag, Boxing Speed Bag)**, **(Breaststroke, Front Crawl)**, (Brushing Teeth, Head Massage), **(Cliff Diving, Diving)**, **(Cricket Bowling, Cricket Shot)**, **(Cutting in Kitchen, Mixing)**, (Drumming, Playing Tabla), (Field Hockey Penalty, Soccer Penalty), **(Hammer Throw, Throw Discus)**, (Hammering, Hammer Throw), **(Handstand Pushups, Handstand Walking)**, **(Head Massage, Haircut)**, **(High Jump, Long Jump)**, **(Horse Race, Horse Riding)**, (Hulahoop, Floor Gymnastics), **(Ice Dancing, Salsa Spin)**, (Javelin Throw, Shotput), (Juggling Balls, Soccer Juggling), **(Jump Rope, Jumping Jack)**, **(Kayaking, Rafting)**, (Knitting, Hulahoop), (Mopping Floor, Cutting in Kitchen), **(Nun chucks, Taichi)**, **(Parallel Bars, Uneven Bars)**, (Pizza Tossing, Mixing), **(Playing Cello, Playing Violin)**, **(Playing Guitar, Playing Sitar)**, **(Playing Daf, Playing Dhol)**, (Playing Flute, Playing Cello), (Playing Piano, Playing Guitar), (Pole Vault, Floor Gymnastics), (Pommel Horse, Balance Beam), **(Pull Ups, Push Ups)**, (Punch, Boxing Punching Bag), **(Rock Climbing Indoor, Rope Climbing)**, (Rowing, Rafting), (Shaving Beard, Haircut), **(Skateboarding, Skiing)**, **(Skijet, Surfing)**, (Skydiving, Cliff Diving), **(Soccer Juggling, Soccer Penalty)**, (Still Rings, Pole Vault), (Sumo Wrestling, Boxing Punching Bag), (Tennis Swing, Table Tennis Shot), (Trampoline Jumping, Jump Rope), **(Typing, Writing on Board)**, **(Walking With a Dog, Biking)**, (YoYo, HulaHoop)

8.2 HMDB51

(Brush Hair, Smile), **(Cartwheel, Flic Flac)**, **(Catch, Throw)**, **(Chew, Eat)**, **(Climb, Climb Stairs)**, **(Dive, Somersault)**, **(Draw Sword, Sword Exercise)**, **(Dribble, Shoot Ball)**, **(Drink, Pour)**, (Fall floor, Climb Stairs), **(Fencing, Sword)**, **(Golf, Swing Baseball)**, (Hand Stand, Cartwheel), (Hit, Kick), **(Hug, Kiss)**, (Jump, Climb), **(Kick, Punch)**, Kick Ball, Shoot Ball, **(Laugh, Smile)**, (Pick, Throw), (Pull Up, Push Up), (Push, Stand), **(Push Up, Sit Up)**, **(Ride Bike, Ride Horse)**, **(Run, Walk)**, (Shake Hands, Hug), **(Shoot Bow, Shoot Gun)**, **(Sit, Stand)**, (Smoke, Drink), (Talk, Turn), **(Turn, Wave)**

8.3 Kinetics100

(Abseiling, Bungee Jumping), **(Air Drumming, Head Banging)**, (Archery, Shotput), (Arm Wrestling, Capoeira), (Barbecuing, Making Pizza), (Belly Dancing, Dancing Macarena), (Bench Pressing, Snatch Weight Lifting), (Biking Through Snow, Skiing), **(Blowing Glass, Blowing Out Candles)**, (Bowling, Golf Putting), (Brushing Teeth, Filling Eyebrows), **(Bungee Jumping, Diving Cliff)**, (Canoeing or Kayaking, Windsurfing), **Capoeira, Tai Chi)**, (Catching or Throwing Frisbee, Throwing Discus), (Cheerleading, Dancing Macarena), **(Climbing Tree, Rock Climbing)**, (Contact Juggling, Spinning Poi), **(Country Line Dancing, Dancing Macarena)**, **(Crawling Baby, Crying)**, (Dancing Ballet, Country Line Dancing), **(Driving Car, Motorcycling)**, **(Dunking Basketball, Playing Basketball)**, **(Dying Hair, Filling Eyebrows)**, (Eating Spaghetti, Making Pizza), **(Feeding Birds, Feeding Goats)**, (Feeding Fish, Feeding Birds), (Flying Kite, Snowkiting), (Golf Putting, Throwing Discus), (Gymnastics Tumbling, Pole Vault), **(Hammer Throw, Throwing Discus)**, **(High Jump, Pole Vault)**, (Hitting Baseball, Golf Putting), **(Hulahooping, Spinning Poi)**, (Ice Skating, Skiing), (Jet Skiing, Wind Surfing), **(Jumping Into Pool, Swimming Backstroke)**, (Marching, Playing Trumpet), **(Milking Cow, Shearing Sheep)**, (Passing American Football, Hitting Baseball), (Mowing Lawn, Climbing Tree), **(Playing Bass Guitar, Playing Guitar)**, **(Playing Cello, Playing Violin)**, (Playing Clarinet, Playing Trombone), **(Playing Harmonica, Playing Recorder)**, **(Playing Harp, Playing Ukulele)**, (Paintball, Skateboarding), **(Playing Squash, Playing Tennis)**, **(Playing Trombone, Playing Trumpet)**, (Playing Ukulele, Playing Guitar), (Presenting Weather Forecast, Reading a Book), (Pull Ups, Snatch Weightlifting), (Pumping Fist, Punching Bag), **(Pushing Car, Pushing Cart)**, (Reading Book, Playing Recorder), **(Riding Elephant, Riding or Walking with Horse)**, **(Salsa Dancing, Tango Dancing)**, **(Scuba Diving, Snorkeling**), (Shot Put, Hammer Throw), **(Side Kick, Punching Bag)**, (Skateboarding, Ice Skating), **(Skiing, Tobogganing**, (Ski Jumping, Tobogganing), **(Snowkiting, Windsurfing)**, (Somersaulting, Capoeira), (Squat, Snatch Weight Lifting), (Washing Dishes, Brushing Teeth), (Yoga, Zumba), (Zumba, Dancing Macarena)

References

1. Gowda, S.N., Rohrbach, M., Sevilla-Lara, L.: Smart frame selection for action recognition. Proceedings of the AAAI Conference on Artificial Intelligence **35**(2), 1451–1459 (May 2021), https://ojs.aaai.org/index.php/AAAI/article/view/16235
2. Gowda, S.N., Sevilla-Lara, L., Kim, K., Keller, F., Rohrbach, M.: A new split for evaluating true zero-shot action recognition. In: 43rd DAGM German Conference on Pattern Recognition (2021)
3. He, K., Zhang, X., Ren, S., Sun, J.: Identity mappings in deep residual networks. In: European conference on computer vision. pp. 630–645. Springer (2016)
4. Jing, L., Parag, T., Wu, Z., Tian, Y., Wang, H.: Videossl: Semi-supervised learning for video classification. In: Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV). pp. 1110–1119 (January 2021)
5. Pagliardini, M., Gupta, P., Jaggi, M.: Unsupervised Learning of Sentence Embeddings using Compositional n-Gram Features. In: NAACL 2018 - Conference of the North American Chapter of the Association for Computational Linguistics (2018)
6. Yun, S., Han, D., Oh, S.J., Chun, S., Choe, J., Yoo, Y.: Cutmix: Regularization strategy to train strong classifiers with localizable features. In: International Conference on Computer Vision (ICCV) (2019)