Few-shot Image Recognition for UAV Sports Cinematography

Emmanouil Patsiouras *Anastasios Tefas* *Ioannis Pitas*

Department of Informatics, Aristotle University of Thessaloniki
{emmanoup, tefas, pitas}@csd.auth.gr

Abstract

The goal of few-shot image learning is to utilize a very small amount of training examples in order to train a machine learning model to recognize a given number of image classes. While humans can perform such a task pretty much effortlessly, applying the same mechanism to deep learning visual recognition systems is a much more difficult task, having a wide range of real-world visual recognition applications. In this paper, we investigate the behavior of such few-shot methods in the context of drone vision cinematography for sports event filming, in order to recognize new image classes by taking into consideration the fact that this new class we wish to identify is a subclass of an already known class. More specifically we use UAV footage to recognize certain types of athletes, belonging to a subset of an original athlete class, utilizing only a handful of recorded images of this athlete subclass. We examine the effects of such methods on image recognition accuracy while proposing a novel approach for accuracy optimizations. The overall task is evaluated on actual cycling race UAV footage.

Index Terms— few-shot learning, image recognition, unmaned aerial vehicles

1. Introduction

Over recent years, we have experienced the vast power of deep learning-based models in several computer vision tasks, notably in image recognition[4] and object detection[16]. However, the superior performance of such models strongly depends on employing a large number, e.g. hundreds or thousands, of training examples for each image class. In many circumstances gathering a sufficient number of training image examples for specific classes can be rather difficult due to data accessibility issues, such as privacy/IPR for facial images. In such cases, training data scarcity can lead to poor generalizations and overfitting, thus significantly limiting the performance of a visual recognition learning model. Moreover, dim efforts for manual gathering and labeling thousands of training examples can be prohibitive, even if crowdsourcing is used. Furthermore, a large number of training samples may lead to many gradient-based training iterations hence imposing further heavy computational costs. Lastly, if a CNN model is already trained for a number of initial given classes and we wish to re-train it to recognize new previously unseen classes a typical procedure is to gather and label a sufficient number of new data for every novel class and start over a new training cycle.

Contrary to the traditional data-hungry training models, few-shot learning methods are typically designed to provide adequate re-training for new classes given a few sample images from each one and primal visual knowledge as extracted from a model already trained on a set of initial classes. For example, in few-shot object recognition, we wish to develop a learning model that is able to accurately recognize and classify unseen objects (meaning new classes) using only 1-5 training examples per new object.

In the past, few-shot learning has been mostly employed and evaluated on some standard few-shot recognition benchmarks, such as Omniglot [5], Mini-ImageNet[14], ImageNet low-shot benchmark[3] and, more recently, Fewshot-CIFAR100[9]. In more recent literature, few-shot learning has been used to explore specific recognition tasks, e.g. pedestrians [15] or micro-organism recognition [11][2]. In this work, however, we investigate the employment of few-shot learning in a different and novel, data-wise, application scenario.

As the employment of camera-equipped UAVs for sport event filming increases rapidly [10][7][8], visual target (in our case athlete) detection, recognition and tracking are very important. In this work, we specifically use few-shot learning techniques on a UAV bicycle race dataset for cyclist detection and tracking. In this context, we have to detect and recognize not only general image classes, e.g. cyclists, runners, boat rowers, but also specific athletes amongst them, e.g. known champions or athletes leading a race. Particularly we apply few-shot learning to recognize the leader of the general ranking of Giro d' Italia cycling race (called "maglia rossa") using only a few samples of

him/her, by re-training a CNN model that recognizes cyclists.

Figure 1. a) Examples of base "cyclist" image class used for CNN training, b) examples of novel "maglia rosa" subclass used for few-shot learning (CNN re-training).

2. Few-shot data recognition scenario and methodology

A very common procedure in addressing few-shot learning (typically referred to as *n-shot k-way* visual tasks) is to combine it with standard deep learning algorithms, in order to optimize classification accuracy of new, previously unseen, object classes. Given abundant training examples for a number of initial classes (from now on called *base* classes), a deep recognition model is trained and is subsequently used for the recognition of *novel* classes, given only a handful of novel class training examples ([12] [13]). In our case, the novel class we wish to recognize ("maglia rosa") is a special subclass of an original base class ("cyclist"), whose main distinguishing characteristic is his/her unique jersey color (pink). Our general base "cyclist" class includes UAV images of cyclists with different backgrounds, brightness, orientations and colors. Representative images of the aforementioned classes are shown in Figure 1.

We follow the methodology described in [1] as it strives to classify any given test sample to either a base class or a novel class in a unified and dynamic manner, by redesigning the classification layer of a typical CNN, in order to include the classification weights of both the base, as learned through a standard CNN training procedure, with many training examples, and the novel classes utilizing only few training examples and (optionally) features learned from the base classes.

In [1], a standard CNN architecture is used, which after four convolution/relu/max-pooling layers, extracts a d-dimensional feature vector $\boldsymbol{z} = F(\boldsymbol{x}|\boldsymbol{\theta}) \in \mathbb{R}^d$ from an input image $\boldsymbol{x}$, given the CNN parameter vector $\boldsymbol{\theta}$ learned during training with base class examples. This feature vector is then fed to a classification layer which computes the final probability classification scores $\boldsymbol{p} = C(\boldsymbol{z}|\mathcal{W})$, where $\mathcal{W} = \{\boldsymbol{w}_k^* \in \mathbb{R}^d\}_{k=1}^{K^*}$ are classification weights vectors, one for every of the K^* object classes. We shall subsequently see that $\boldsymbol{w}_k^*$ could come from either a base class or a novel one.

In [1], authors instead of using the typical dot-product in order to calculate the raw probability classification scores p_k, the cosine similarity operator is used:

$$p_k = \tau \cdot \overline{\boldsymbol{z}}^T \overline{\boldsymbol{w}}_k^*, \tag{1}$$

where $\overline{\boldsymbol{z}}$ and $\overline{\boldsymbol{w}}_k^*$ are the l_2-normalized vectors $\boldsymbol{z}, \boldsymbol{w}_k^*$, respectively and τ is a learnable scalar value for adjusting the cosine similarity range to fit the softmax function domain. This modification, along, with our novel contribution (to be subsequently described), is proven to be very crucial for accurate base class and novel subclass recognition in our visual application task.

An initial form of the $\mathcal{W}$ classification weights for the base classes are generated from a first, standard CNN training cycle given a dataset with sufficient training examples for every base class. : $\mathcal{D}_{train} = \bigcup_{k=1}^{K} \{\boldsymbol{x}_{k,i}\}_{i=1}^{N_b}$, where K is the number of bases classes, N_b is the number of training examples of the k-th class and $\boldsymbol{x}_{k,i}$ is its i-th training example. For the classification weights of a novel class a meta-parameter generation mechanism is used [1] that gets as input: a) the feature vectors $\boldsymbol{Z}' = \{\boldsymbol{z}_i' = F(\boldsymbol{x}_i'|\boldsymbol{\theta})\}_{i=1}^{N'}$ of the N' examples of that novel class (typically $N' \leq 5$) and b) the initial form of $\mathcal{W}$, to generate a new classification weight $\boldsymbol{w}'$ for that novel class. Accumulating the inferred classification weights for all the novel K' classes in a set $\mathcal{W}' = \{\boldsymbol{w}_k'\}_{k=1}^{K'}$ and setting $\mathcal{W} = \mathcal{W} \cup \mathcal{W}'$ in the CNN classification layer, the model is enabled to recognize both base and novel classes.

A simple first thought for inferring the classification weights for a novel category is to average the normalized feature vectors of the few training examples $\boldsymbol{z}_i'$ [1]:

$$\boldsymbol{w}' = \boldsymbol{w}_{avg}' = \frac{1}{N'} \sum_{i=1}^{N'} \overline{\boldsymbol{z}}_i' \tag{2}$$

However, this does not fully exploit the information acquired from CNN training on the base classes as it does not include the visual information present in their classification weights plus the existence of a limited number of training images. Instead, the aforementioned average mechanism is enhanced with an attention-based mechanism that allows

the feature vectors $\boldsymbol{z}_i, i = 1, ..., N'$ to exploit the base classification weights that are most similar to them.

$$\boldsymbol{w}'_{att} = \frac{1}{N} \sum_{i=1}^{N'} \sum_{k=1}^{K} f(\overline{\boldsymbol{z}}'_i, \overline{\boldsymbol{w}}_k) \cdot \overline{\boldsymbol{w}}_k \tag{3}$$

where $f(\cdot)$ is a cosine similarity function followed by a softmax operation that quantifies similarity to each base classification weight $\boldsymbol{w}_k, k = 1, ..., K$. The final classification weight vector is:

$$\boldsymbol{w}' = \boldsymbol{\theta}_{avg} \odot \boldsymbol{w}'_{avg} + \boldsymbol{\theta}_{att} \odot \boldsymbol{w}'_{att} \tag{4}$$

where $\odot$ is the elementwise multiplication operator and $\boldsymbol{\theta}_{avg}, \boldsymbol{\theta}_{att} \in R^d$ are learnable weight vectors of a second training stage, using the same dataset for the base classes (as in the first training stage).

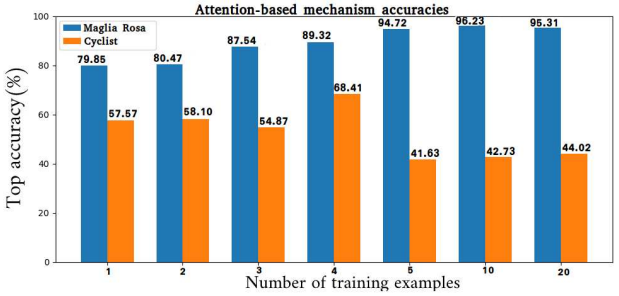
We found, by experiments that deploying the above inference mechanisms for generating the classification weights of a novel class, doesn't perform well in cases where the novel class we wish to recognize is a subclass of a base class, as it happens in our application scenario. In that case, the above model tends to favor the recognition of the novel ("maglia rosa") class over the base ("cyclist") class hence failing to provide adequate generalizations for our desired classes.

Therefore we explored the geometrical structure of the base class, by partitioning the training examples of our desired base class into a C number of clusters using k-means [6]. We then feed the training examples of each of these clusters as a separate base class to our CNN and start over a completely new training cycle. In this case our new training dataset is: $\mathcal{D}_{train} = \bigcup_{k=1}^{K+C} \{\boldsymbol{x}_{k,i}\}_{i=1}^{N_b}$, where C is the number of clusters. This modification is explained by the fact that we wish our novel subclass to exploit as much information as possible from not only one base class (as in the previous discussion) but from C distinguished base classes that are similar to each other. To the best of our knowledge, this approach is novel in the context of few-shot learning. We call this novel approach Cluster Few-Shot (CFS) classification.

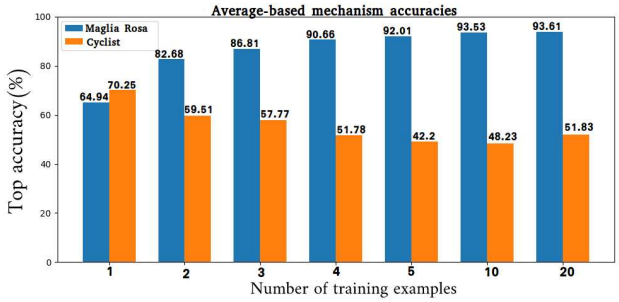
3. Experimental study

3.1 Baseline Few-Shot Classification

For our empirical evaluation, state-of-the-art [1] is used as a testbed for implementation and for performance comparison. We specificly chose this method because it is one of the few from the related literature that during the evaluation step it compares test samples from the novel learned classes in the combined set of basic and novel classes. We initially train our model in a number of base classes, namely the Mini-ImageNet training subset of a total of 64 classes [14] containing 600 training examples, of size 84×84,



(a)



(b)

Figure 2. Average classification accuracies of [1] on both base "cyclist" and novel "maglia rosa" classes using a) average and b) attention mechanism vs the number of training examples for the novel class.

for each image class. Furthermore, we enhance the above dataset with the extra class "cyclist" (not present in the original dataset) using the same number of cyclist images as for every other class in the original dataset, in order to subsequently facilitate the recognition of the "maglia rosa" cyclist subclass using very few training examples. We compute the classification weights for both the 65 base classes and one novel subclass as described in the previous section, without incorporating the clustering mechanism. For the validation stage, we feed our model with images of both "cyclist" and "maglia rosa" classes and we measure the recognition accuracy solely on them. At this point, it is very important to mention that we wish our model to perform satisfactorily on both "cyclist" and "maglia rosa" classes. Figure 2 shows the average classification accuracies using both the average (2) and attention (4) mechanism for a varying number of "maglia rosa" training examples. For the base "cyclist" class, the classification weights remain the same in all cases. It can be observed that the use of the attention-based mechanism produces, in general, slightly improved recognition accuracy on the novel "maglia rosa" class. In practice, this means that the method indeed takes into consideration the information extracted from the base class. However, although we have obtained great results in the recognition of our novel "maglia rosa" class the corresponding results on the base "cyclist" class are rather poor in comparison and

reduce when the number of the training novel "maglia rosa" examples increases.

3.2 Cluster Few-Shot Classification

In this stage, we incorporate the clustering mechanism we described in the preview section. During testing, if a sample belonging to the original "cyclist" base class is assigned to any one of the newly formed base clusters, it is assumed to belong to the general base "cyclist" class. Figure 3 shows the average classification accuracies of our novel CFS classification method using the attention-based mechanism and partitioning our initial class into 3, 5 and 10 clusters, respectively. The number of clusters was arbitrarily selected to have a small value and stadily increase. As can be seen, CFS has slightly reduced the performance on the novel "maglia rosa" class but vastly improved the performance on the base "cyclist" class leaning towards a more balanced recognition model.

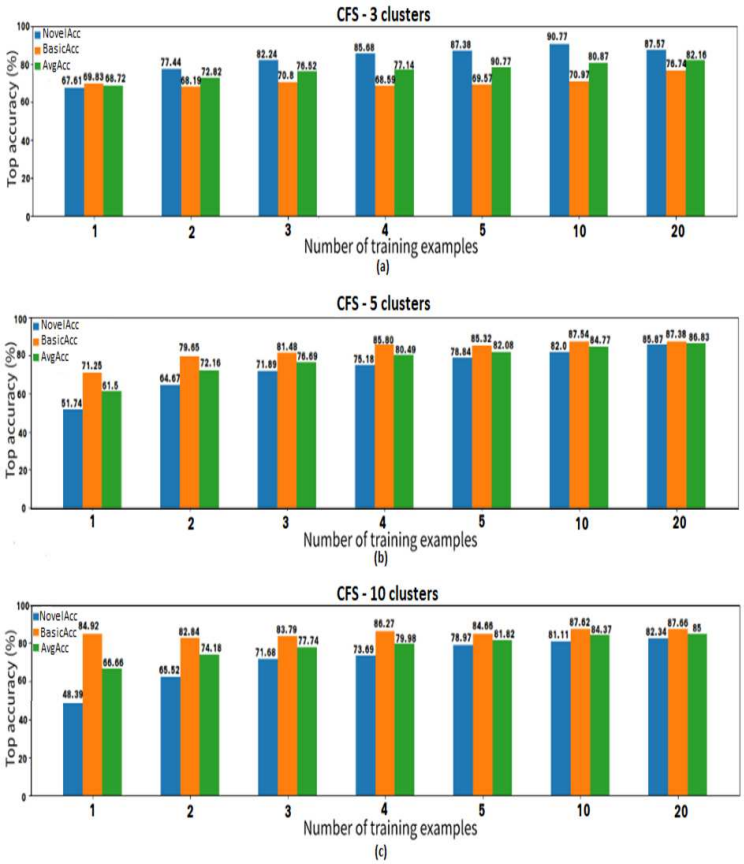


Figure 3. Average classification accuracies of CFS method on both base "cyclist" and novel "maglia rosa" class using attention mechanism for a) 3 clusters, b) 5 clusters and c) 10 clusters vs the number of training examples for the novel class.

4. Conclusions

In this paper, we have discussed some practical difficulties involving the training procedure of some typical data-hungry training methods. We have talked about how few-shot learning methods can help lift these bottlenecks and provide adequate generalizations on certain image recognition tasks. We have applied few-shot learning techniques to recognize certain types of athlete classes, subset of an original image class, in the context of UAV sports cinematography. We have proposed a novel approach for handling these types of recognition tasks and we have shown, through our experiments, how our novel approach can optimize recognition accuracy.

Acknowledgment

This work has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No731667(MULTIDRONE). This publication reflects the author's views only. The European Commission is not responsible for any use that may be made of the information it contains.

References

[1] S. Gidaris and N. Komodakis. Dynamic few-shot visual learning without forgetting. *Proceedings of the IEEE Conference of Computer Vision and Pattern Recognition*, pages 4367–4375, 2018.

[2] S. Gull and F. Minhas. Amp0: Species-specific prediction of anti-microbial peptides using zero and few shot learning. *arXiv preprint arXiv:1911.06106*, 2019.

[3] B. Hariharan and R. Girshick. Low-shot visual recognition by shrinking and hallunicating features. *Proceeding of the IEEE International Conference on Computer Vision*, pages 3018–3027, 2017.

[4] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. *Proceedings of the IEEE Conference of Computer Vision and Pattern Recognition*, pages 770–778, 2016.

[5] B. Lake, R. Salakhutdinov, J. Gross, and J. Tenenbaum. One shot learning of simple visual concepts. *Proceedings of the Annual Meeting of the Cognitive Science Society*, pages 301–304, 2018.

[6] S.P Lloyd. Least squares quantization in pcm. *IEEE Transactions on Information Theory*, pages 129–137, 1982.

[7] P. Nousi, I. Mademlis, I.Karakostas, A. Tefas, and I. Pitas. Joint lightweight object tracking and detection for unmaned vehicles. *Proceedings of the IEEE International Conference on Image Processing*, pages 160–164, 2019.

[8] P. Nousi, E. Patsiouras, A. Tefas, and I. Pitas. Convolutional neural networks for visual information analysis with limited computing resources. *Proceedings of the IEEE International Conference on Image Processing*, pages 321–325, 2018.

[9] B. N. Oreshkin, P. Rodríquez, and A. Lacoste. Tadam: task dependent adaptive metric for improved few-shot learning. *NeurIPS*, pages 721–731, 2018.

[10] F. Patrona, I. Mademlis, A.Tefas, and I.Pitas. Computational uav cinematography for intelligent shooting based on semantic visual analysis. *Proceedings of the IEEE International Conference on Image Processing*, pages 4155–4159, 2019.

[11] S.M. Schröder, R. Kiko, and R. Koch J.O. Irisson. Low-shot learning of plankton categories. *German Conference on Pattern Recognition*, pages 301–304, 2018.

[12] J. Snell, K. Swersky, and R. Zemel. Prototypical networks for few-shot learning. *NeurIPS*, pages 4077–4087, 2017.

[13] Q. Sun, Y. Liu, T. S. Chua, and B. Schiele. Meta-transfer learning for few-shot learning. *Conference of Computer Vision and Pattern Recognition*, pages 3630–3638, 2016.

[14] O. Vinyals, C. Blundell, T. Lillicrap, and D. Wierstra. Matching networks for one shot learning. *NeurIPS*, pages 3630–3638, 2016.

[15] L. Xiang, X. Jin, J. Han G. Ding, and L. Li. Incremental few-shot learning for pedestrian attribute recognition. *International Joint Conference on Artificial Intelligence*, pages 3912–3918, 2019.

[16] H. Xu, C. Jiang, X. Liang, L. Lin, and Z. Li. Reasoning-rcnn: Unifying adaptive global reasoning into large-scale object detection. *Proceedings of the IEEE Conference of Computer Vision and Pattern Recognition*, pages 6419–6428, 2019.