Deep Image Classification of a Wild Data Set for Olympic Sports

Daniel Ferreira Moreira[1], **Cristina Nader Vasconcelos**[1], **Aline Paes**[1], **Luis Velho**[2]

[1]Instituto de Computação – Universidade Federal Fluminese (UFF)
Caixa Postal 24210-346 Niterói – RJ – Brazil

[2]Instituto Nacional de Matemática Pura e Aplicada (IMPA) Rio de Janeiro – RJ – Brazil

dmoreira@ic.uff.br, crisnv@ic.uff.br, alinepaes@ic.uff.br, lvelho@impa.br

Abstract. *Automatic retrieval of information is better achieved when the data is previously classified in categories. However, given the amount of currently available information, and being constantly produced, it is very hard to manually classify all of them. Thus, it is essential to establish a methodology that would be able to not only retrieving data, but also to present them in its correct class. One way to accomplish this later goal is to use Machine Learning algorithms for classification. In this work we tackle this problem using a Deep Learning technique, given the increasingly success of this area in classification tasks. Particularly, we address the automatic classification of images related to the Olympic Games, focusing on predicting which sport a image is associated with. We present an extensive empirical study encompassing the training the Deep Learning Network known as GoogLeNet with 26 of the 42 Olympic sports disciplines and 5362 images. From the results, we show that GoogLeNet is indeed able to correctly classify most of the images that were not presented to the network in the training phase.*

1. Introduction

Image classification with Machine Learning algorithms is still a challenging task due to the numerous and complex features that a single picture usually contains. However, over the internet and other sources of information, the usage of images is greatly adopted, and with so, it is essential to use such algorithms for image classification in tasks such as summarization and searching.

With the Rio 2016 Olympic Games, a great number of images of the games were shared throughout the web, ranging from traditional media covering sites to social media such as *Facebook* or *Twitter*. Since the Rio 2016 Olympic Games had 42 Olympic sport disciplines, it can be convenient in a number of situations to automatically retrieve only a small subset of all these related pictures presented on the internet, as if you were looking for only one sport, for example.

Over the years, Convolutional Neural Networks (CNNs) [Krizhevsky et al. 2012] have achieved the state of the art results for image classification. Motivated by such results, we investigate the performance of GoogLeNet [Szegedy et al. 2015b], a recently developed CNN topology, for the task of classifying a novel data set presented in this work. To do so, we extracted thousands of images from Google search engine results for

the previous Olympic games and annotate them accordingly to their respective Olympic sport discipline.

This rest of this paper is organized as follows. Section 2 discuss related work focused on the classification of sports related content. Section 3 briefly explains the GoogLeNet architecture. Section 4 details the process adopted for the creation of the Olympic Sports Data Set, and also the enhanced variations of it that were made for the propose of investigating the performance of the neural network. Section 5 presents the result and discussions about them in terms of accuracy for the classification process. We end the paper in Section 6, with the final concluding remarks and possible future work.

2. Related Work

In this section we briefly present previous works that also addressed the task of image and video classification for sports related content, encompassing both traditional Machine Learning algorithms and the more recently developed Deep Learning techniques. In the first case, there must be an initial step to collect the relevant features from the images, that in turn are going to be the input of the Machine Learning method. In the second case, on the other hand, both the features and the classifier are automatically extracted from the learning process.

In [Jung et al. 2004] a traditional Machine Learning pipeline is followed, where first features from color histogram, color coherence vector, edge direction histogram and edge direction coherence vector are extracted from the images. Then, such features are used as the attributes of the image data set. Those are going to be the random variables considered by the classifier to learn patterns from the images. Next, a Bayesian Classifier [Friedman et al. 1997] is trained to automatically assign a class to a sport image.

While the later mentioned work addressed a discriminative Machine Learning algorithm, in [Jeon and Kim 2015], a Latent Dirichlet allocation (LDA) is adopted, that is able to produce a generative probabilistic model of a corpus. The LDA [Blei et al. 2003], extended with spatial information, was able to classify the images contained in UIUC's Sports Event Data Set[Li and Fei-Fei 2007].

Concerning the application of a Deep Learning method, in [Karpathy et al. 2014] a data set of one million YouTube sports videos is used to train a Convolutional Neural Network [Krizhevsky et al. 2012]. The output of the network is a layer with 487 different classes, arranged in a hierarchic manner, for example the data set contains 6 different types of bowling, 7 different types of American football and 23 types of billiards.

Previous to that, in [Breen et al. 2002] a shallow neural network following a domain-dependent ontology was trained to perform classification also in the sports domain. After classifying the objects within the image into predefined classes, the ontology is traversed in order to find a possible semantic relationship among those objects.

This paper also presents a study of classification over a dataset on the sports domain. And motivated by a lack of available sports datasets for the Olympic games we decided to create our own. Since results using neural networks seemed promising, it was also the methodology applied.

3. GoogLeNet

The Convolutional Neural Networks(CNN) provided a new approach to image classification. CNNs are able to learn hierarchies of invariant features and classifiers from annotated data sets.

They currently represent the state of the art solution for classification problems over natural images. Since its 2012 edition, this is the case of the ImageNet Large Scale Visual Recognition annual Challenge (ILSVRC) [Russakovsky et al. 2015] for object localization/detection and image/scene classification from images and videos at large scale, in which participants received 10 million annotated images and are supposed to classify its context into 1000 classes.

Typically, different CNNs topologies combine neurons that are sensitive to small sub-regions of the visual field, and able to learn a hierarchy of visual features composed by stacks of linear (convolutional) and non-linear transformations, that add invariance to the transformation process. The convolutional layers work like stacks of classical Computer Vision filter banks, except that in this case such filters are learned from the CNN training process.

Usually, the convolutional layers of a CNN are followed by fully connected layers, similar to classical neural networks, and as such they have the role of a classifier over the feature vector generated by the previous layers.

The topology named as GoogLeNet[Szegedy et al. 2015a] was proposed to tackle the problem of image classification with a very deep neural network architecture and is the ImageNet Large Scale Visual Recognition Competition 2014 (ILSVRC) [Russakovsky et al. 2015] winner.

Its main contribution was the development of the Inception Module (Figure 1) that dramatically reduced the number of parameters in the network, creating a deeper and wider topology. The input signal that enters an inception module is processed by filters of different sizes, including 1×1 filters that are responsible for aggregating correlated features, while in previous networks the convolution kernels were typically uniform.

GoogLeNet is formed with Inception modules(1) stacked on the top of each other, leading to a 22-layer deep model when only taking account layers with parameters, and 27 layers when including pooling, and the overall number of layers used for the construction of the network is about 100. Along with the convolution layers it also has non-linear transformations such as Rectified Linear Units (ReLU) and pooling layers [Szegedy et al. 2015a].

Differing from the previous CNN approaches, the GoogLeNet topology replaces the fully connected layers by sparse ones. In this way, the classifier decision is spread throughout the network, even inside the convolutions.

In order to combat the vanishing gradient problem while keeping the network deep, during training, they included two auxiliary classifiers connected to the intermediate layers of the network, so that the their loss gets added to the total loss of the network with a discount weigh, but that at discarded during inference time.

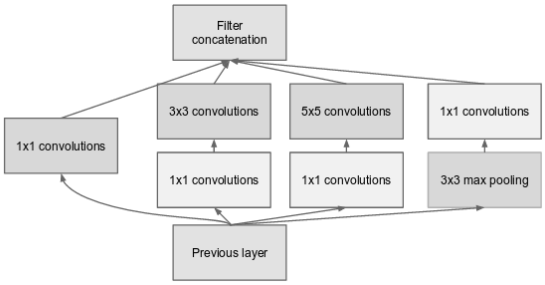

Figure 1. Inception Module [Szegedy et al. 2015a]

4. Wild Dataset for Olympic Sports

In order to provide a rich data set to train the GoogLeNet, we gathered a number of images from the web using Google's Custom Search Engine tool. As those images were captured under uncontrolled conditions and devices we call it the Wild Data Set of Olympic Sports, in other words, those images were not cherry-picked neither manipulated, with the only exception being the removal of logos and drawings that did not correspond to real world pictures. We select the twenty-six more popular Olympic sports out of the forty-two played on Rio 2016, namely: Archery, Athletics, Badminton, Basketball, Beach Volleyball, Boxing, Canoe Sprint, Cycling, Equestrian, Fencing, Field Hockey, Football, Golf, Artistic Gymnastics, Handball, Judo, Rhythmic Gymnastics, Rowing, Sailing, Shooting, Swimming, Synchronized Swimming, Tennis, Volleyball, Water Polo and Weightlifting. We also included 2 additional classes to represent the images related to the Olympic Torch and to the Medal Ceremonies. Taking everything into account, the final data set contains 5362 images, of which 3763 are used to training the weight parameters of the network, 802 are used to validate the training and 797 are left out of the whole training phase, in order to compose the prediction test set. This data set is henceforth mentioned as the *raw data set*, and bellow on Table 1 is a class-by-class division showing the amount of samples that populates each one of the training, validation and test sets.

In addition to those original images, a few steps of*data augmentation* were implemented to artificially increase the data set in order to have a greater number of varied images to increase the generalization of the network. This becomes relevant since unlike the original ILSVRC challenge where the GoogLeNet was first trained from, which contained about 1.2 million images[Russakovsky et al. 2015], our data set has a much lower number, so artificially increasing this number can prove to be useful. This process consists of creating new images by performing flips, translations, rotations and cropping on the original images. Thus, the images were horizontally and vertically flipped and rotated up to 45 degrees either to the left or to the right. Cropping and translating were done in conjunction in a way that square images of at least 200×200 pixels were created from the original images. These steps were followed only to the 3763 images reserved for training in the *raw data set* in a way that the validation and test sets remain the same. Then, the resulting training set consists of 22522 images. The same 802 validation images and 797 test images mentioned before for the *raw data set* are used here. This data set is henceforth mentioned as the *augmented data set*.

The original (raw) data set has a different number of samples for each class. As a consequence, when augmenting the data set to produce the second data set, this one

Table 1. Class-by-class division for Training, Validation and Testing

	Train	Val	Test
archery	152	32	32
athletics	153	33	33
badminton	155	33	33
basketball	147	32	32
beach volleyball	72	16	16
boxing	198	42	42
canoe	160	32	34
cycling	63	14	14
equestrian	160	34	34
fencing	180	39	39
field hockey	142	30	30
football	154	33	32
golf	176	38	38
gymnastics	156	33	33
handball	134	28	26
judo	180	38	38
medals	53	11	11
rhythmic gymnastics	68	15	15
rowing	168	36	36
sailing	68	14	14
shooting	61	13	13
swimming	194	41	39
synchronized swimming	64	14	14
tennis	143	31	31
torch	68	15	15
volleyball	171	36	34
water polo	167	35	35
weightlifting	156	34	34

remains imbalanced. It's worth mentioning that on challenges such as ILSVRC, this is also not a problem since the data set used is already balanced from scratch. The ideal situation is to always have the same number of images per category, so that when training the network, it does not favor one class in detriment of the others. Thus, aiming at addressing such problems that commonly arise from highly skewed data [Chawla 2009], we created a third stratified balanced data set. As more images typically make the networks learn better, we decide to perform oversampling instead of undersampling, to balance the data. Thus, we randomly included duplicated samples for all classes, except the most populated one, of the *augmented data set*, in a way that each one is going to have the same number of images as the most populated class on the training set.

The most populated class on the augmented data set is the *boxing* class, with a total sum of 1188 images. Therefore, for each class, random copies of the images were added in a way that they ended up with 1188 images each. For example the *fencing* class had 1080 images, so 108 copied images were added to this class. This process yielded 33200 training images. As before, the same 802 images are used for validation and 797 for testing. From now on, this data set will be mentioned as the *balanced data set*.

5. Experimental Results

Data sets Besides the data sets created in this work and explained in Section 4, we also experiment on the UIUC's Sports Event Data Set[Li and Fei-Fei 2007], to compare a previous developed sports data set with the ones yielded in the current work. Although it does not have the same classes as the Olympics data set, we can use it to observe how well the GoogLeNet can generalize its prediction along different, albeit related, domain. As usual when employing a deep learning technique, we start from the public available previous optimized weights for the GoogLeNet topology, learned from the ILSVC data set, and fine-tuned them to our domain. However, in order to observe the behavior of the network when its weights are learned from scratch, we also present results when the GoogLeNet were trained from random weights (without the fine-tuning) with the*raw* and *augmented* data sets.

Experimental Methodology For each test described below, it was used the GoogLeNet neural network using an image size of 256×256 pixels, 30 training epochs with a snapshot and validation interval of 1 epoch and a batch size of 8 images. The solver type used is the Adaptive Gradient(AdaGrad) [Duchi et al. 2011] and the base learning rate adopted was 0.001 with a Sigmoid Decay policy, with 50% step and gamma 0.1. These are the default parameters for training this network and it was not a concern of the present work to optimize them.

Results Table 2 shows the accuracy results for training both with and without fine-tuning, regarding the raw data set. It shows both the overall accuracy and the top 5 accuracy. From the results, we can observe that the performance of the network is greatly improved when we take advantage of previously learned weights, and only adjust them during the training phase. In addition, we can see that the accuracy results are very close for both the validation and test sets, which is an indicative that the network is generalizing well the domain.

Table 2. Accuracy for the Olympic data set

	Raw	**Raw(fine-tune)**	**Augmented**	**Balanced**
Acc.(val.)	0.6262	0.8873	0.8997	0.9158
Acc.(test.)	0.6161	0.8733	0.8683	0.8745
Top5 Acc.(val.)	0.9047	0.9801	0.9826	0.9851
Top5 Acc.(test.)	0.8921	0.9749	0.9686	0.9787

In order to understand the performance of GoogLeNet for this type of classification, the same experiment was performed on a different data set that also has images related to the sports genre. The data set used was the UIUC's Sports Event Data Set[Li and Fei-Fei 2007]. The data set contains 1579 images spread throughout 8 classes: rowing, croquet, badminton, rock climbing, snowboarding, sailing, polo and bocce. Using the same training parameters as the previous experiment, the obtained accuracies for validation and testing were 0.7708 and 0.7542 respectively and the top 5 accuracy for validation and test were 0.9791 and 1 respectively.

Table 2 also shows the accuracy obtained from both the augmented set and balanced set, considering these modified data to train the network, but the same validation and test set images, to verify the performance of the network. To further understand the classification done on each training, it was calculated a confusion matrix for the test set. By taking for each class the amount of correct and incorrect predictions, it is possible to find an individual accuracy for each class. The individual accuracy for each class obtained on each of the 3 sets trained from the data produced in this work are listed on Table 3.

Table 3. On the left: Individual accuracy for each class, considering the training with the data sets created in this work. The bold value is the best one for each sport discipline. On the right: Acc. with 0.9 threshold.

	No threshold			0.9 threshold		
	Raw*	**Aug.**	**Bal.**	**Raw***	**Aug.**	**Bal.**
archery	0.9063	**1**	0.9688	1	1	1
athletics	**0.9091**	0.8788	0.8788	0.9655	0.9642	0.9333
badminton	0.7576	**0.7879**	**0.7879**	0.8275	0.8125	0.8333
basketball	**0.9063**	0.7813	0.8438	0.9583	0.8888	0.9259
beach volleyball	1	1	1	1	1	1
boxing	0.9048	0.9048	**0.9286**	0.9743	0.9047	0.9750
canoe	0.7647	0.7941	**0.8235**	0.9230	0.8227	0.8571
cycling	**1**	**1**	0.9286	1	1	1
equestrian	**0.9412**	0.8529	0.8824	1	0.9062	0.9677
fencing	0.9744	0.9744	**1**	1	0.9062	0.9677
field hockey	**0.8333**	0.8000	0.7333	0.9585	0.8846	0.9545
football	**1**	0.9688	0.9688	1	1	1
golf	0.9474	**1**	0.9737	1	1	1
gymnastics	0.9697	0.9091	**1**	0.9696	0.9677	1
handball	**0.8462**	0.7308	0.8077	0.8947	0.8095	0.9047
judo	0.9474	0.9474	**0.9737**	1	0.9722	0.9729
medals	0.5455	**0.6364**	0.4545	1	0.7500	0.7142
rhythmic gymnastics	0.8667	0.8667	**0.9333**	0.8333	0.9166	1
rowing	0.9167	0.9167	**0.9444**	0.9293	1	0.9705
sailing	1	1	1	1	1	1
shooting	0.9231	0.9231	0.9231	1	1	1
swimming	0.6923	**0.7179**	0.6410	0.7575	0.8181	0.6756
synchronized swimming	0.8571	0.8571	**0.9286**	0.9230	1	1
tennis	0.5484	0.5161	**0.5806**	0.7647	0.6000	0.5454
torch	**0.8667**	0.8000	0.8000	1	0.9166	0.8571
volleyball	0.6765	**0.8000**	0.7059	0.8400	0.7500	0.8214
water polo	0.9143	0.7353	**0.9429**	0.9666	0.9705	0.9687
weightlifting	**1**	0.9714	**1**	1	1	1

:corresponding to fine-tuned column from Table 2

By carefully analyzing the results presented on Table 3, its possible to see that only a few classes reach a very poor accuracy performance on the test set, some getting around 0.5 accuracy only. The augmented improved on 7 classes and got worse on 12, and the balanced improved on 12 classes and got worse on 12 also. Figure 2 shows a example of a photo that improved its classification by adding the balance to the dataset in relation to the augmented, on the augmented set the top prediction was of 0.4963 for basketball, on the balanced dataset the top prediction was the correctly predicted handball with 0.999.

Figure 2. Classified as basketball with 0.4963 using the augmented set and as handball with 0.999 using the balanced dataset

Figure 3. First is a tennis image wrongly classified as field hockey. Second is a badminton image wrongly classified as tennis

The *medals* class is the extreme example on the case of low performance on the training with the balanced set and reaching a cap of only 0.6364 on the training with the augmented set. One possible reason for this behavior is the fact that the class itself is very ambiguous, since it contains images that could be perceived as any other sport instead of only the medal celebration for that particular sport. This can be easily noted when looking at the confusion matrix for the balanced training, where the medal images classified as something else are well spread throughout other classes. When analyzing each image individually its possible to find cases where the top5 prediction contains the medal class as its first prediction with $0.95+$ and some other cases where the first prediction is something else with $0.95+$ and the medal class does not even show up on the first five predictions.

A surprisingly poor result can also be seen for the *tennis* class for all 3 training, scoring somewhere around the 0.5 mark for each one. When jumping closely to the confusion matrix its possible to see that the network often treats a *tennis* image as *volleyball*, *badminton* or more commonly *field hockey*. Figure 3 shows a case where the balanced network tags a tennis image as field hockey with 0.9989 followed by a case where the same network tags a badminton image as tennis.

Next on the list of undesired performances on classification is the *swimming* class, which seems to be confused with other 2 classes: *water polo* and *synchronized swimming*. This is understandable, as they are all aquatic sports. Even though, those 2 classes have a good individual performance.

On some cases, where it was expected to find poor results we actually found, surprisingly a very good accuracy score. A good example is the rowing and canoe classes which can be easily confusing for the human eye to differentiate. Figure 4 shows a example where the balanced network tagged both sports with accuracy of 1.

Figure 4. On the top a canoe image. On the bottom a rowing image

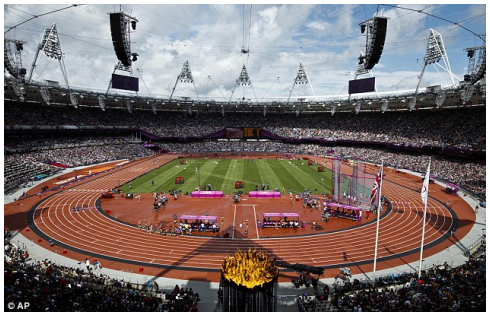

Figure 5. First prediction basketball, second prediction athletics

Sometimes, when the network makes a mistake, it is possible that the reason was a conflict between two classes with similar probabilities. For example: figure 5 have on its top5 prediction something as the first one being basketball with 0.5016 and the second one being athletics with 0.4982. The correct answer would be athletics but the answer the network gives is basketball. However, these values seem to have no statistical difference. By adding a certain threshold to the classification and getting only predictions with a certain confidence determined by this threshold its possible to see how well the network performs without those cases. By only taking images classified with a prediction equals or higher than 0.90, different results can be observed, as exhibited in Table 3.

These results show that the raw approach does indeed have more trouble with competing predictions as it has a bigger accuracy percentage when such threshold is applied. Although this seems to give a better accuracy for picking isolated images from a given class, this cannot be seen as a metric for classification, since some images are left out when using a threshold. Also its possible to see that even though the overall accuracy is higher, some classes remains *problematic*, such as *tennis* and *medals*.

6. Conclusion

In this paper we showed that a Deep Neural Network can achieve a high performance on image classification for a novel data set of the Olympics sports domain. Furthermore, we show that a few steps of image processing on the data set can improve significantly the results of difficult classes in terms of accuracy.

Although some sports can be easily perceived by the network, others seem to need a bigger attention during the training phase. Classes like swimming and tennis got very low accuracy scores and it might be possible to further investigate if some image processing techniques can better explicit those unique features that these sports have when training the network.

References

Blei, D. M., Ng, A. Y., and Jordan, M. I. (2003). Latent dirichlet allocation. *Journal of machine Learning research*, 3(Jan):993–1022.

Breen, C., Khan, L., and Ponnusamy, A. (2002). Image classification using neural networks and ontologies. In *Database and Expert Systems Applications, 2002. Proceedings. 13th International Workshop on*, pages 98–102. IEEE.

Chawla, N. V. (2009). Data mining for imbalanced datasets: An overview. In *Data mining and knowledge discovery handbook*, pages 875–886. Springer.

Duchi, J., Hazan, E., and Singer, Y. (2011). Adaptive subgradient methods for online learning and stochastic optimization. *Journal of Machine Learning Research*, 12(Jul):2121–2159.

Friedman, N., Geiger, D., and Goldszmidt, M. (1997). Bayesian network classifiers. *Machine learning*, 29(2-3):131–163.

Jeon, J. and Kim, M. (2015). A spatial class lda model for classification of sports scene images. In *Image Processing (ICIP), 2015 IEEE International Conference on*, pages 4649–4653. IEEE.

Jung, Y., Hwang, E., and Kim, W. (2004). Sports image classification through bayesian classifier. In *Current Topics in Artificial Intelligence*, pages 546–555. Springer.

Karpathy, A., Toderici, G., Shetty, S., Leung, T., Sukthankar, R., and Fei-Fei, L. (2014). Large-scale video classification with convolutional neural networks. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, pages 1725–1732.

Krizhevsky, A., Sutskever, I., and Hinton, G. E. (2012). Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pages 1097–1105.

Li, L.-J. and Fei-Fei, L. (2007). What, where and who? classifying events by scene and object recognition. In *2007 IEEE 11th International Conference on Computer Vision*, pages 1–8. IEEE.

Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M., Berg, A. C., and Fei-Fei, L. (2015). ImageNet Large Scale Visual Recognition Challenge. *International Journal of Computer Vision (IJCV)*, 115(3):211–252.

Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V., and Rabinovich, A. (2015a). Going deeper with convolutions. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–9.

Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J., and Wojna, Z. (2015b). Rethinking the inception architecture for computer vision. *arXiv preprint arXiv:1512.00567*.