A Machine Learning Approach to assess the Interlocutor Attention: from Data Collection to Real-Time Application

Simone Tedeschi
Sapienza University of Rome
tedeschi.1762897@studenti.uniroma1.it

Sveva Pepe
Sapienza University of Rome
pepe.1743997@studenti.uniroma1.it

Abstract

Many human-computer or human-robot interactions require the capability of the system of understanting whether the user is paying attention or not. However, to train such systems, large amounts of data are needed, but they are currently unavailable. In this paper, we first address the issue of data scarcity by creating a large dataset – with about 120k images – for the attention detection task. Then, we develop a strong baseline system which is able to correctly perform the task, achieving competitive results on the proposed dataset. Further to the above considerations, we extend our system by: i) adding an auxiliary face detection module, and ii) introducing a novel GAN-based data augmentation technique. Finally, we design a web application to enable real-time testing of the developed model. We released the collected data and software – code and model checkpoints – at https://github.com/sted97/sted97.github.io/.

1 Introduction

Human-Robot Interaction (HRI) (Goodrich and Schultz, 2008), or more generally Human-Computer Interaction (HCI) (Karray et al., 2008), gained an increasing interest in the last two decades thanks to recent technological advances in the hardware and systems fields (Balakrishnan et al., 2015). Several studies have been put forward with the aim of improving the quality of the interactions – allowing the use of voice commands or gestures – by exploiting modern machine learning approaches (Trigueiros et al., 2012; Tandel et al., 2020). A good part of such interactions are based on visual perception, and would require the capability of the system of understanding whether the user lost the focus or not, but few approaches, which rely on eye-gaze, have been proposed (Macrae et al., 2002; Yu and Smith, 2013).

Attention detection is a crucial step in various tasks and applications. For instance, when dealing with children affected by hyperactivity disorder, understanding whether the child is following or not the proposed activity (e.g., an educational lesson) is needed, in the latter case, to propose again the same activity, or pause it. Furthermore, in a standard HRI interaction, if the user is not heedful when the robot asks a question (e.g., "What do you want to order?", in a restaurant scenario), it can turn up the volume and repeat the question to grab attention. Finally, in a driver monitoring system, checking whether the driver is not looking ahead can be useful to alert him, and bring his attention back. Nevertheless, to train deep learning systems – usually in a supervised fashion – able to solve such tasks, massive amounts of data, which are currently not available, are required.

In this paper, we focus on the attention detection task in its entirety, from addressing the issue of data paucity, passing through the creation of a baseline system and the proposal of novel strategies to further improve performances, up to the development of a web application. More specifically, our main contributions can be summarized as follows:

- We introduced a new manually-annotated dataset for the attention detection the task, containing about 120k images from 18 different users;
- We developed a strong baseline system for the task, which is able to achieve competitive results on the proposed dataset;
- We enhanced our baseline system by adding an auxiliary face detection module, achieving a boost in performances;
- We proposed a novel GAN-based data augmentation technique to further enrich the collected data, again attaining performance improvements;
- We extensively evaluated the benefits of our

contributions on the proposed test set by performing a statistical analysis on the obtained results;

- We designed a web application to further analyze the behavior of the system and enabled real-time testing.

We hope that our work will encourage further studies on the development of high-performance attention detection systems. We release data and software – code and model checkpoints – at https://github.com/sted97/sted97.github.io/.

2 Data Collection Process

With recent advances in neural networks (Gu et al., 2017), a large variety of tasks can be successfully solved by training such systems in a supervised manner on large amounts of data. However, for certain tasks, such data is a scarce or unavailable resource, hence, it need to be collected. For this purpose, we created a new large dataset for the attention detection task.

Usually, in a general interaction – between two humans or between a human and a machine – one of the most important features for understanding whether the interlocutor is heedful or not is its head direction. In particular, if the interlocutor is looking frontally towards its interaction partner, it can be assumed,with a high probability, that he is paying attention. On the other hand, if he is looking in another direction (e.g., left, right, up or down) it can be assumed that he is distracted by something else in the environment (e.g., its smartphone). We followed the above described intuition and designed our dataset in the following way:

- we use five classes – CENTER, LEFT, RIGHT, UP and DOWN – rather than a binary label, to better cluster different situations, letting the model distinguish between various kinds of inattention. The CENTER class is the only positive label for our task, indicating that the interlocutor is heedful;

- we recorded 270 videos from 18 different users, each video lasting ∼20 seconds. In particular, each user was asked to record 15[1] videos, changing its location and/or its outfit – including glasses – every 5 videos, to let the dataset be as general as possible;

[1] each user recorded 3 series of 5 videos, where each of the 5 videos corresponds to one specific label.

Dataset Split	Class	# Samples
Train	CENTER	16.5K
	LEFT	17.8K
	RIGHT	18.0K
	UP	17.3K
	DOWN	17.0K
Validation	CENTER	2.5K
	LEFT	2.7K
	RIGHT	2.6K
	UP	2.4K
	DOWN	2.6K
Test	CENTER	3.3K
	LEFT	3.4K
	RIGHT	3.4K
	UP	3.7K
	DOWN	3.3K
$\sum$	—	116K

Table 1: Dataset statistics describing the number of samples for each class in each of the three dataset splits.

- we segmented these videos and we manually double-checked the validity of the annotations pre-assigned by the users.

We provide our dataset with the standard split in train, validation and test sets. The three datasets are disjoint. In particular, the people involved in the train set, namely 13, are not present in the validation and test sets, which contain 2 and 3 persons respectively. Also the vice-versa holds. On average, the people characterizing our dataset belong to the 20-30 years-old age range. However, we added a person with a completely different age, namely ∼60 years-old, to test the capability of our system to generalize across different ages. Statistics about the above described dataset are provided in Table 1.

3 Attention Detection System

In this Section we first describe our baseline system (Section 3.1). Then we will discuss a couple of extensions to improve such system (Sections 3.2 and 3.3).

3.1 Baseline System

One of the most effective techniques developed in the last few years is transfer learning (Pan and Yang, 2009). This technique consists in exploiting the knowledge of models trained to address one specific task as a starting point for another task.

Usually, the aim of such pre-trained models is to be as general as possible to enable fine-tuning for a large variety of other tasks.

In particular, we make use of the Visual Geometry Group 16 (VGG16) (Simonyan and Zisserman, 2015) pretrained model, selected among other alternatives (details in Section 4), to address our attention detection task. It's a Convolutional Neural Network (CNN) consisting of 13 convolutional layers – separated by 5 pooling layers – and 3 final dense layers, on top of which we added two additional dense layers – of 1024 and 128 cells respectively – and an output layer with softmax activation function.

The final objective of our system is to be able to generalize as much as possible, being robust to noisy or unseen samples. For this purpose, a common technique is to augment training data (Shorten and Khoshgoftaar, 2019), but in our case not all data augmentation techniques can be applied. For instance, *horizontal flipping* cannot be applied because left labeled images would become right-labeled, and vice-versa, creating confusion in the training process. On the other hand, *brightness* and *shifting* can be applied. The former is beneficial when the test set contains samples with brightness levels completely different from the ones available in the training data. Instead, the latter is useful when the position of the interlocutor can significantly vary between the training and test instances. However, since this feature is already intrinsically present in our training set, *shifting* does not improve performances in our case. Further details are provided in Section 4.

3.2 Face Detection

Another fundamental task of computer vision is face detection (Zafeiriou et al., 2015), whose aim is to identify the human face relying on the key points that characterize it, namely eyes, mouth, nose, etc. It can be used as an auxiliary step for a wide variety of tasks, for instance, it is useful for understanding facial expressions (Matsugu et al., 2003), to allow lip reading (Lucey and Potamianos, 2006), and in marketing applications (Ishii et al., 2002).

In particular, we exploit face detection to let our system focus only on the relevant features for our task. For instance, when dealing with low-contrast images – images where the background color is very similar to the face color, or where an excessive exposure is present – it can be very challenging for our attention detection system to distinguish between different classes. To remedy this problem, we use an auxiliary face detection system[2] which outputs the four vertices of the bounding box surrounding the face, and we use them to crop the original image. An example of application of the face detection system is shown in Figure 1.

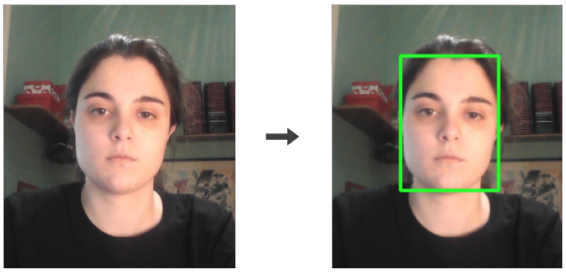

Figure 1: Example of application of the face detection system.

3.3 GAN-based Data Augmentation

As we mentioned in Section 3.1, data augmentation techniques are beneficial for several computer vision tasks, enabling the development of more powerful systems. However, such techniques consist in elementary mathematical operations, like *shifting*, *rotating* and *flipping* images, which in some cases are not enough.

In this Section, we introduce a novel GAN-based data augmentation technique which is able to further improve system performances. In particular, this strategy consists in generating new samples, starting from the images of the users available in the training set and making them with a different age. For this purpose, we adopt SAM (Alaluf et al., 2021), a C-GAN for face aging, which takes as input an image of a person P and a desired age a_d, and outputs the transformed image of P with a_d.

This strategy is fundamental in any computer vision application which deals with users from all ages, but usually, the corresponding training set does not cover all the needed ages. Indeed, in our case, only users with ages between 20 and 30 years are included, but our system is intended to be used indistinctly both young and old people. To overcome this issue, in addition to the data augmentation techniques already described in Section 3.1, we apply this face aging solution to make our system insensitive to age changes, as illustrated in Figure 2.

[2]We consider a pre-trained OpenCV Caffe Model as face detection system.

Figure 2: Example of face aging with target age $a_d = 60$.

Hyperparameter	Value
batch size	64
learning rate	1e-5
dropout	0.5
adam β_1	0.9
adam β_2	0.999
adam ϵ	1e-8

Table 2: Hyperparameter values of the models used for our experiments.

4 Experiments

In this Section, we first describe our experimental setup (Section 4.1), then we discuss the results of our baseline system (Section 4.2.1), and finally we present the results of our complete baseline + face detection system (Section 4.2.2) and of our GAN-based contribution (Section 4.2.3).

4.1 Experimental Setup

We implemented our baseline system and its extensions with Tensorflow (Abadi et al., 2016), using the Keras framework[3] to load and fine-tune the weights of pretrained models. We trained each model configuration for 30 epochs, adopting an early stopping strategy with a patience value of 5, with Adam (Kingma and Ba, 2017) and a learning rate of 10^{-5} and a cross-entropy loss criterion. In the remainder of this Section, we report the results of the best model checkpoints according to their accuracy on the validation split of the dataset, described in Section 2, at the end of each training epoch. The complete list of hyperparameter values is provided in Table 2. Furthermore, for the results shown in Sections 4.2.2 and 4.2.3, we repeat each training on 5 different seeds, fixed across experiments, and report the mean and standard deviation of their accuracy score; we compare experiments by means of Student's t-test (Student, 1908).

[3] https://github.com/keras-team/keras

Model	Accuracy
ResNet152V2	53.37
ResNet50	54.74
Xception	49.95
VGG16	**74.21**
VGG19	74.12

Table 3: Accuracy of the different models on our test set.

Model	Accuracy
VGG16	74.21
w/ *shift*	74.32
w/ *brightness*	**75.11**
w/ *brightness* + *shift*	74.95

Table 4: Performances of the different data augmentation techniques when applied to the VGG16 model.

4.2 Results

4.2.1 Baseline System

Our first aim is to select a strong baseline system to solve our attention detection task. For this purpose, we perform model selection on the validation set comparing different architectures. The obtained results are shown in Table 3.

As we can see from the table, the ResNet152V2, ResNet50 (He et al., 2016) and Xception (Chollet, 2017) models behave badly due to their high architectural complexity – they have 152, 50 and 36 layers respectively – suggesting that extremely deep architectures are not the most suitable choice for our task. On the other hand, simpler architectures such as VGG16 and VGG19 (Simonyan and Zisserman, 2015) perform much better. We chose VGG16 as our temporary baseline, on which we applied data augmentation techniques, because it attains comparable performances with respect to the 19-layers version, while being even less complex.

At this point, in order to further improve our system, we applied the data augmentation techniques illustrated in Section 3.1. We reported the results in Table 4. As can be seen from these results, applying the shifting operation does not increase system performances, being already partially included in our dataset. Instead, brightness, as already mentioned in Section 3.1, improve performances, especially because our test set contains samples with highly variable levels of brightness. In the rest of our discussion, we refer to this final system, VGG16 with brightness, as *Baseline System*.

Model	Accuracy	Δ
Baseline System	73.31 ± 1.40	–
w/ *Face Detection*	86.17 ± 1.66**	12.85
w/ *Face Detection* + *GAN*	**88.45** ± **1.11***	2.28

Table 5: Performances of the *Face Detection* and *GAN* contributions when applied to the *Baseline System*. ** stands for $p < 0.001$, * stands for $p < 0.05$. Both statistical significance and differences (Δs) are always expressed with respect to the immediately above row.

4.2.2 Face Detection

As we mentioned in the Section 3.2, we start from the intuition that the background of the image, or the outfit of the user, do not provide any useful information, but can only introduce noise in the classification process. Therefore, we perform face detection as an auxiliary task to let the system focus only on the relevant features. This extension produce a three-fold advantage: i) it significantly increases the performances, ii) it dramatically reduces training times, and iii) it considerably decreases the amount of required disk space.

Regarding system performances, as it can be seen from the second line of Table 5, the complete *Baseline System* + *Face Detection* model obtains an average improvement of 12.85 accuracy points over the previously described *Baseline System*. Moreover, we observe an extreme statistical significance in our results, having a p-value of 3.7e-07, showing how this technique is fundamental for our task. In Figure 3, we can also see how the validation accuracy of the two systems vary during epochs, observing a consistent gap between the two curves (the pink and grey ones).

Considering training times instead, while the *Baseline System* requires ∼35 minutes for each epoch, pre-processing the images by applying face detection, and training the system directly on the cropped images, we observe a speed-up of ∼6x.

Finally, the last advantage consists in the reduced amount of required storage space. The standard dataset occupies ∼25GB of memory, whereas the one pre-processed using face detection requires ∼4GB only.

4.2.3 GAN-based Data Augmentation

As already discussed in Section 2, our training set contains only images of users belonging to the 20-30 years age range. Nevertheless, about the 30% of our test set is made by samples from a user of about 60 years. While standard data augmentation techniques are not sufficient to let the system generalize well with users with different ages, our GAN-based data augmentation technique, illustrated in Section 3.3, helps to mitigate this issue. In particular, as we can see from Table 5, enhancing our system using GANs, provides further improvements over a yet very strong system. Once again, we found statistical significance in our results.

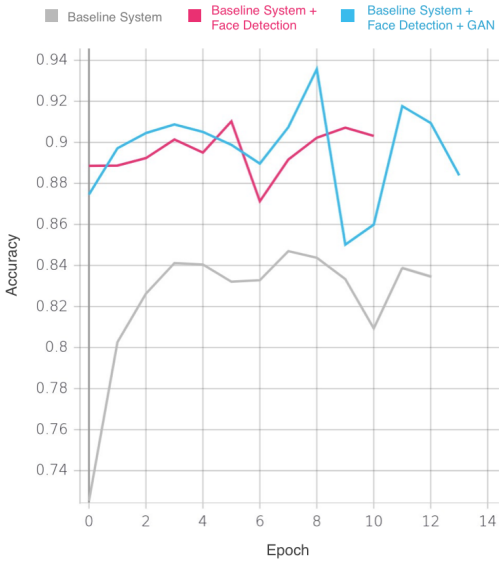

Figure 3: Validation performances of *Baseline System*, *Baseline System* + *Face Detection* and *Baseline System* + *Face Detection* + *GAN*.

5 Real-time Application

After we collected our dataset (Section 2), developed our *Baseline System* (Section 3.1), and applied our extensions – face detection (Section 3.2) and GAN-based data augmentation (Section 3.3) – our final objective was to realize a real-time application to further analyze and test our system.

Our application is made by two pages. The first one is descriptive, and illustrates the contents of our work and its possible use cases. Instead, the second one is the actual application, where it is possible to play with our system and check its predictions. An example of positive and negative predictions is shown in Figure 4. We implemented our application using HTML, CSS, JavaScript, JQuery and TensorflowJS[4], and it works on any browser and device, but it is not optimized for smartphone and tablet. We made it publicly available at the following url: https://sted97.github.io/.

[4] https://www.tensorflow.org/js

Figure 4: Example of positive and negative predictions in our web application.

6 Conclusions and Future Work

Although attention detection is a crucial step in various computer vision tasks and applications, in particular in the HRI field, few works aim at solving this task. In this paper instead, we addressed the attention detection task in its entirety.

We started by collecting a new manually-annotated dataset, consisting of ∼120k images from 18 different users, with 5 possible labels (Section 2). Then, we exploited such dataset to train a baseline system and to meausure its performances, obtaining satisfactory results on the produced test set (Section 3.1).

Further, as a first extension, we proposed a larger system which includes an auxiliary face detection module to remove useless informations for the task (e.g., the background), obtaining consistent performance improvements over the baseline system (Section 3.2). As a second extension instead, since our system is intended to be suitable for users of all ages, we proposed in Section 3.3 a novel GAN-based data augmentation technique for face aging in order to let our system being robust to age changes. This extension produced further performance improvements over a yet very strong system. We also extensively evaluated the benefits of our contributions on the proposed test set by performing a statistical analysis, finding statistical significance in our results (Section 4.2).

Finally, to further analyze the behavior of the developed attention detection system and enable real-time testing, we designed a web application (Section 5).

As future work we plan to extend it to multi-face attention detection.

References

Martín Abadi, Paul Barham, Jianmin Chen, Zhifeng Chen, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Geoffrey Irving, Michael Isard, et al. 2016. Tensorflow: A system for large-scale machine learning. In *12th {USENIX} symposium on operating systems design and implementation ({OSDI} 16)*, pages 265–283.

Yuval Alaluf, Or Patashnik, and Daniel Cohen-Or. 2021. Only a matter of style: Age transformation using a style-based regression model.

Nikilesh Balakrishnan, Thomas Bytheway, Lucian Carata, Oliver R. A. Chick, James Snee, Sherif Akoush, Ripduman Sohan, Margo Seltzer, and Andy Hopper. 2015. Recent advances in computer architecture: The opportunities and challenges for provenance. In *Proceedings of the 7th USENIX Conference on Theory and Practice of Provenance*, TaPP'15, page 8, USA. USENIX Association.

Francois Chollet. 2017. Xception: Deep learning with depthwise separable convolutions. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Michael A Goodrich and Alan C Schultz. 2008. *Human-robot interaction: a survey*. Now Publishers Inc.

Jiuxiang Gu, Zhenhua Wang, Jason Kuen, Lianyang Ma, Amir Shahroudy, Bing Shuai, Ting Liu, Xingxing Wang, Li Wang, Gang Wang, Jianfei Cai, and Tsuhan Chen. 2017. Recent advances in convolutional neural networks.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Yohei Ishii, Hitoshi Hongo, Makoto Kanagawa, Yoshinori Niwa, and Kazuhiko Yamamoto. 2002. Detection of attention behavior for marketing information system. In *7th International Conference on Control, Automation, Robotics and Vision, 2002. ICARCV 2002.*, volume 2, pages 710–715. IEEE.

Fakhreddine Karray, Milad Alemzadeh, Jamil Abou Saleh, and Mo Nours Arab. 2008. Human-computer interaction: Overview on state of the art.

Diederik P. Kingma and Jimmy Ba. 2017. Adam: A method for stochastic optimization.

Patrick Lucey and Gerasimos Potamianos. 2006. Lipreading using profile versus frontal views. In *2006 IEEE Workshop on Multimedia Signal Processing*, pages 24–28. IEEE.

C Neil Macrae, Bruce M Hood, Alan B Milne, Angela C Rowe, and Malia F Mason. 2002. Are you looking at me? eye gaze and person perception. *Psychological science*, 13(5):460–464.

Masakazu Matsugu, Katsuhiko Mori, Yusuke Mitari, and Yuji Kaneda. 2003. Subject independent facial expression recognition with robust face detection using a convolutional neural network. *Neural Networks*, 16(5-6):555–559.

Sinno Jialin Pan and Qiang Yang. 2009. A survey on transfer learning. *IEEE Transactions on knowledge and data engineering*, 22(10):1345–1359.

Connor Shorten and Taghi M Khoshgoftaar. 2019. A survey on image data augmentation for deep learning. *Journal of Big Data*, 6(1):1–48.

Karen Simonyan and Andrew Zisserman. 2015. Very deep convolutional networks for large-scale image recognition.

Student. 1908. The probable error of a mean. *Biometrika*, 6(1):1–25.

Nishtha H Tandel, Harshadkumar B Prajapati, and Vipul K Dabhi. 2020. Voice recognition and voice comparison using machine learning techniques: A survey. In *2020 6th International Conference on Advanced Computing and Communication Systems (ICACCS)*, pages 459–465. IEEE.

Paulo Trigueiros, Fernando Ribeiro, and Luis Paulo Reis. 2012. A comparison of machine learning algorithms applied to hand gesture recognition. In *7th Iberian conference on information systems and technologies (CISTI 2012)*, pages 1–6. IEEE.

Chen Yu and Linda B Smith. 2013. Joint attention without gaze following: Human infants and their parents coordinate visual attention to objects through eye-hand coordination. *PloS one*, 8(11):e79659.

Stefanos Zafeiriou, Cha Zhang, and Zhengyou Zhang. 2015. A survey on face detection in the wild: past, present and future. *Computer Vision and Image Understanding*, 138:1–24.