An investigation of transfer learning for deep architectures in group activity recognition

Karl Casserfelt
Dept. of Computer Science, Malmö University,
Malmö, Sweden
karl.casserfelt@gmail.com

Radu-Casian Mihailescu
Dept. of Computer Science, Malmö University
Internet of Things and People Research Center
Malmö, Sweden
radu.c.mihailescu@mau.se

***Abstract*—Pervasive technologies permeating our immediate surroundings provide a wide variety of means for sensing and actuating in our environment, having a great potential to impact the way we live, but also how we work. In this paper, we address the problem of activity recognition in office environments, as a means for inferring contextual information in order to automatically and proactively assists people in their daily activities. To this end we employ state-of-the-art image processing techniques and evaluate their capabilities in a real-world setup.**

Traditional machine learning is characterized by instances where both the training and test data share the same distribution. When this is not the case, the performance of the learned model is deteriorated. However, often times, the data is expensive or difficult to collect and label. It is therefore important to develop techniques that are able to make the best possible use of existing data sets from related domains, relative to the target domain. To this end, we further investigate in this work transfer learning techniques in deep learning architectures for the task of activity recognition in office settings. We provide herein a solution model that attains a 94% accuracy under the right conditions.

***Index Terms*—Computer Science; Machine learning; Activity recognition.**

I. Introduction

The field of computer vision, where the goal is to allow computer systems to interpret and understand image data, has seen in recent years great advances with the emergence of deep learning. Deep learning, has been shown to be the state-of-the-art technique to solve the problem of object recognition in image data e.g. [11], [12], [16]. One of the next big challenges in computer vision is to allow computers to not only recognize objects, but also activities. This study is an exploration of the capabilities of deep learning for the specific problem area of activity recognition in office environments

The area of activity recognition in office environments is rather complex as a computer vision classification problem. In related areas such as object recognition, the goal is to identify distinct objects in images. Activity recognition on the other hand is normally performed on data where subjects are moving and interacting. Scene recognition aims to detect and classify the scenes of images. When it comes to activity recognition for office scenarios, relevant information about what is going on can potentially be a rather complex mixture of the three mentioned categories of visual classification. In order to infer whether an activity is for example a presentation seminar, it is not enough to identify the activities of individual users but also the combination of activities and interaction between them. At the same time, objects can discern and separate activities from each other. If a person uses a computer it means something else than if they are using a whiteboard. Even scene recognition and context plays some part in this problem space as a group activity could mean different things if it takes place in a conference room or a lounge area.

The workplace environment is being impacted by technological innovations in a significant way, especially with the recent advent of the internet of things [9]. This transformation is largely being referred to as the transition to a smart office, which fosters agile and flexible work. Thus, instead of being static and passive, the environment adapts and contextualizes the experience to the needs and preferences of its users. This can range from controlling things like heating, lighting, or ventilation in order to increase user satisfaction ([8], [21]), to more complex scenarios, where a dynamic set of devices, with their functionalities and services, cooperate temporarily to achieve a user goal ([2], [10]). Hence, detecting activities in office environment has applications to a large number of use cases, where the extracted context can be communicated to various subsystems that could proactively assist office users or maintenance personal with their daily activities.

However, especially in the case of image data sets, the training data is often expensive or difficult to collect and label. Consequently, there is a clear need to reuse and repurpose existing data sets for new tasks and domains. This approach is generally referred to as transfer learning, which is the ability of a system to recognize and apply knowledge and skills learned in previous tasks to novel tasks or new domains. Specifically, in this work we set out to investigate the extent to which deep learning architectures, that prove to have a high performance in solving image recognition problems, could benefit from applying transfer learning for activity recognition in office space environments.

II. Related work

Several proposals have achieved well functioning activity recognition algorithms using deep learning and non-visual data, such as motion sensor data. Yang et. al [20] achieved promising results using body mounted motion sensors and deep learning, and displayed a robust model without the need for hand-crafted feature extraction. A similar approach was

978-1-5386-9151-9/19/$31.00 ©2019 IEEE

taken by Ronao and Cho [13], who instead of dedicated sensors, used smart phones to collect data. Apart from the obvious difference that none of the two mentioned proposals used visual data, a key difference from this work is that they focused on single person activity recognition with no regard for scene or interaction. However, there are some aspects that are similar to our work. Namely, Yang et. al [20] achieved robustness and removed the need for hand-crafted features, which is similar to the goal of using raw video data as an input. Ronao and Cho [13] on the other hand went for a more unobtrusive design, which again is one of the benefits of the proposed solution of this paper.

In [5], the authors achieved state-of-the-art performance in group activity recognition in video data through the use of a hierarchical model that combines a number of techniques. The authors make use of a tracklet software that first locates and isolates the people in a frame. The areas of the frame that contain people are ran through a convolutional network in order to extract image features, which are then passed through a long short term memory cell (LSTM), one for each person. This is to retain temporal features and relative changes for individual subjects. At the same time, features that relate to the group of subjects as a whole are used in another LSTM, and the process is repeated in a two step-method. The solution achieves an accuracy score of 81.5% when tested on the Collective Activity Dataset [3]. A benchmark comparison was presented by fine-tuning the AlexNet pre-trained model and classifying each frame by itself without temporal features, which achieved a 63% accuracy. This shows that the authors significantly improved the scores by introducing their hierarchical model, where temporal features were used for both individuals and group separately.

However, the problem space in [5] differs from our setting in the type of data they are concerned with. Both the Collective Activity Dataset [3] and the authors' own sports-dataset are picturing activity classes that relate to a rather high physical mobility. Compare this to most activities in an office setting, such as meetings, presentations or silent work. These type of classes most often imply that very few of the subjects in a video move from their seat. Furthermore, their solution presupposes that the tracklet software for isolating people works as expected and robustly, which may not always be the case for scenarios where users sit down, are immobile or have their back turned to the camera. For instance, many methods of finding people in frames are based on background subtraction [19], which relies on people constantly moving in the frame.

Another work that combines spatial and temporal features in activity recognition tasks is proposed by Karpathy et. al. [6], who uses a combination of convolutional and recurrent neural networks (RNNs) to classify activity in the Sports-101 dataset, containing a large number of sports videos. They found that the accuracy results that they achieved using spatial features in classifying the videos frame-by-frame were 59.3%. Surprisingly, the inclusion of temporal features using a RNN only resulted in a very small performance increase, reaching 60.9% accuracy, and their best result achieved 63.9% accuracy. These results are interesting because it is intuitive that a better result for video classification should be reached with spatio-temporal features. In comparison with [5], it does seem like the type of data and pre-processing that is used has great significance for the effectiveness of temporal features.

To sum up, it appears that previous state-of-the-art attempts to address group activity recognition using video data and deep learning have a few factors in common. It seems like convolutional neural networks are used as a foundation in most successful implementations in the field, and they are usually combined with recurrent neural networks to retain temporal features. However, the effectiveness of incorporated temporal features are not guaranteed to yield large performance boosts. Furthermore, both the level of pre-processing and type of data can create variations in accuracy, and it does seem like there are not yet any suggested best practice to approach this problem.

III. Proposed approach

In this section we explain in detail the steps of our proposed approach. First, a set of controlled experiments will be performed on an existing data set in a controlled environment to investigate which deep learning model and configuration performs best on the given problem of group activity recognition. Then, a new data set will be constructed from real world activity in a smart office environment and evaluated. To this end, we will investigate the extent to which we can exploit transfer learning in the context of deep learning models and provide a comparative analysis against models that are built from scratch.

A. Controlled Experiments

The first stage of the process will use a modified subset of the data from the AMI Meeting Corpus dataset [4]. The AMI Meeting Corpus dataset is a set of recordings from meeting rooms. It includes a large amount of different types of collected data from each meeting session, including video data collected with up to six different cameras. Despite that a number of camera angles were present for each session, only those that had an overview of the entire room were selected to be incorporated in the used dataset. While this data set contains video streams from meeting rooms and office environment, it does not contain all of the activity classes that are indented to be used by our model. What it does offer instead, is very controlled environments where meetings, presentations and no activity (empty room) are taking place in the exact same conditions, with the same camera angles, lighting, and even participants.

For the purpose of our experiments, all video files that we considered were hand annotated to correspond to one of three classes: *presentation*, *meeting*, or *empty*. The reason for selecting these three classes as an initial set of actions for training comes from the fact that the video files used in the experiments usually had all three of these classes present in the same meeting session. This meant that after annotation

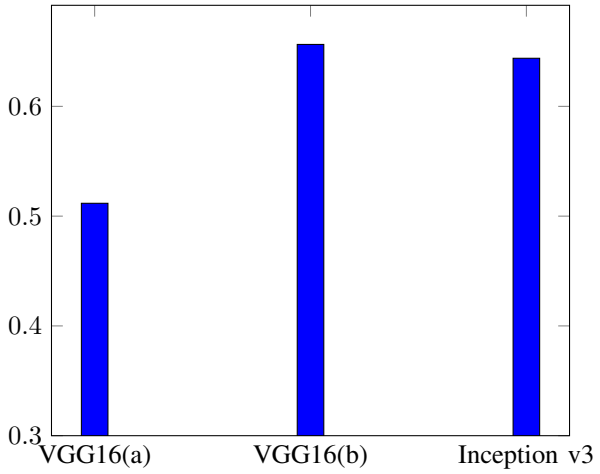

Fig. 1. Accuracy results for:. VGG16(a) using the VGG16 model with random weights; VGG16(b) using the same model with pre-trained weights from the Imagenet dataset; and Inception V3 using the Inception model with pre-trained Imagenet weights.

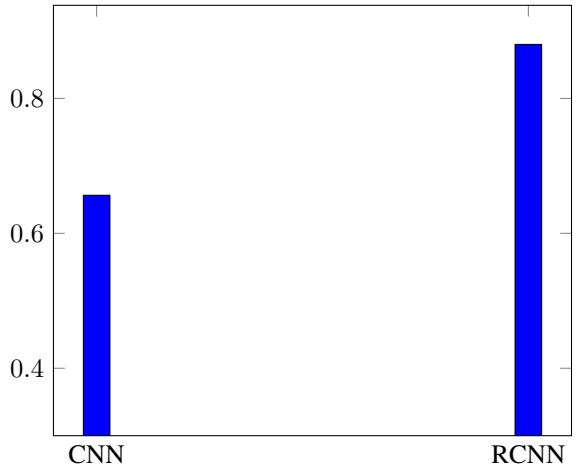

Fig. 2. Accuracy scores of transfer learning on VGG16 using normal CNN technique, and with an RCNN model which included an LSTM layer.

and splitting the data, there would be instances of each class where all conditions of the video were exactly the same. The camera angle, scene, lighting conditions and even people are the same, but their activity differs. This fact would ideally aid in avoiding over-fitting the learning model to conditions other than the activity.

The distinction between *meetings* and *presentations* was defined as follows. A presentation is considered to occur when all the subjects direct their attention to one person, who is physically separated from the rest. This could be a person standing by a whiteboard, projector or other object, while the rest sit down. A meeting on the other hand is when all subjects have similar poses and physical location (most often sitting at the same table), and have their focus directed to the group.

This data set is used to perform initial experiments to find the best base model for the problem space. As will be evident later, this is a rather challenging data set and it is expected that if a model handles it well it could be extendable to more data and classes. The reason for why no data from other sources are used at this stage is to avoid over-fitting the model to unrelated conditions and thus creating unreliable or misleading results.

B. Real World Testing

When a model has been selected given the results of the controlled experiments, the model will be applied to a novel data set captured in the IoTaP Lab in Malmö University[1]. The data set is captured from two different cameras in the lab, recording for seven weekdays, and then manually labeled and sorted into activities. The final model will be trained and validated using this data to review its performance in a real-world scenario.

[1]https://iotap.mau.se

IV. Experimental Results

In this section we report on a number of experiments done on the modified AMI Meeting Corpus data. The videos are processed in the way that each video frame is extracted into a separate JPEG image and resized to 224x224 pixels. Each video is also copied once and flipped horizontally, creating a mirror image of the original. This was done in order to expand the total volume of data without having to use exact copies of previous data. For experiments where no temporal features were included, and each image was treated as its own discrete data point, random transformations were made during training such as zoom, tilt and crop, again to increase data size and decrease over-fitting. All meeting and presentation videos are exactly five minutes long, while the empty class videos have a varying, but shorter length. The videos were split into 25170 images, and 15% of each class were dedicated to a test set. Images belonging to the same videos are kept together, again to avoid over-fitting.

A. Transfer Learning vs. Novel Training

Despite the fact that transfer learning, where pre-trained neural networks are employed and fine tuned in order to solve new problems, are widely considered to be a practical way to leverage previous efforts to reduce computational cost in deep learning tasks, their benefits are not always guaranteed. We hypothesize that this work will benefit from leveraging transfer learning, but at the same time there are no widely available pre-trained networks to use that address a similar problem or that are trained on similar data. The expectation is that by using a general object detection network as a starting point, it's pre-trained capabilities of detecting mid-level features such as edges and shapes will aid in classifying the frames of the AMI Meeting Corpus videos.

For this purpose, both the VGG16 [17] and Inception Model V3 [18] will be used as a foundation for transfer learning in this problem. While there are many opportunities to tweak the learning process using this technique, for example by

specifying what layers are allowed to update their values in the original models, or building various models on top of the pre-trained layers, this stage will aim to achieve a baseline score of the overall performance of the technique and the different models. For this reason, all of the layers of the original models are locked from updating their weights and only a dense 3-layer model will be built on top of the last non-fully connected layer of the original models. Note that these experiments do not take temporal features into account, and thus it treats each video frame as its own image with no relation to the rest of the data.

Three experiments were conducted: *(i)* one where the VGG16 model was used, and its weights were randomly initialized, *(ii)* one where the VGG16 model was used but the weights were pre-trained on the Imagenet dataset [14], and *(iii)* finally the Inception V3 model with pre-trained Imagenet weights. This allows for comparison between transfer learning and novel training, and between the VGG16 and the Inception models. For each experiment we allowed the models to train for one hour on a GTX 1060 video card.

As shown in Fig. 1, the pre-trained models outperformed the model that started with random weights by approximately 14-15% The VGG16 model with random weights achieved a 51% accuracy, while the same model with pre-trained weights achieved 65.5%. However, there was a small difference between a pre-trained VGG16 model and a pre-trained Inception V3 model, as the Inception V3 model achieved 64.4 % accuracy, only 0.9% below the pre-trained VGG16 model.

A separate test was performed using the VGG16 model with pre-trained weights, but excluding the class *empty*, so that it became a binary classification problem between the classes *meeting* and *presentation*. The reason for this is that the two classes are much more similar to each other than the class *empty*, and in the worst case scenario, the accuracy score could have been achieved by randomly classifying between the *meeting* and *presentation* class, while classifying the *empty* class with high accuracy, thus still achieving a good overall score. The result from this test showed an accuracy score of 67.1%, which disproves the above hypothesis.

B. Temporal Features vs. Only Spatial

In this set of experiments we focus on evaluating the added value of incorporating temporal features. The reason behind this experiment is that the activities depicted here are rather inactive in nature, and it does not seem entirely clear that video sequences would provide a better understanding of the occurring activity.

Here were set out to investigate the immediate effects of incorporating RNN elements to the models. For the first experiment, a model was built identical to the best scoring model from the previous section, namely a pre-trained VGG16 model, by removing the top layers, while adding one input layer, two dense layers and one dropout layer. The difference in this experiment is that a LSTM layer is added right after the input to the model. As suspected, this resulted in a much higher accuracy of 88%, outperforming the corresponding model without temporal features with almost 11 percentage points (see Figure 2). This goes on to show that temporal features indeed provide value to this problem space, and even produces a substantial improvement. A summary of the results is given in Table I.

TABLE I
OVERVIEW OF RESULTS FROM EXPERIMENTS COMPARING VGG16 AND INCEPTION V3, PRE-TRAINED WEIGHTS AND RANDOM WEIGHTS, TWO CLASS AND THREE CLASS, AND RCNN WITH LSTM USING CONTINUOUS AND DISCRETE CLASSIFICATION

Base model	LSTM	Weights	Setting	Score
VGG16	No	Random		**51.0%**
Inception V3	No	Pre trained		**64.4%**
VGG16	No	Pre trained		**64.5%**
VGG16	No	Pre trained	'Empty' class exluded	**67.1%**
VGG16	Yes	Pre trained	Continous classification	**52.1%**
VGG16	Yes	Pre trained	Discrete classification	**88.0%**

C. 3D Convolutional Network

Results from the previous section strongly suggests that despite the perhaps mild movement done in office scenarios, the movement that does occur provides means for much better distinction between activities and that temporal features are vital for successful classification. As RNN's with LSTM cells are not the only means of including temporal features in visual data neural networks, it does seem viable to explore more options for temporal feature inclusion. For this reason, a 3D Convolutional Network was implemented and the data was reprocessed to fit this kind of model.

In particular, 3D ConvNets treat sequences of images as three dimensional objects where the images are stacked on each other and the convolutions are performed on the now three dimensional block of pixels. These objects have a width and a height as a normal image, but the depth represents the different frames of the video separated in the time domain. This can lead to very heavy computational tasks that require high-end hardware if the input 3D images are not re-scaled in all dimensions. In this experiment, each image was scaled to a width and height of 32 pixels, and a depth of 10 images in order to handle the task in a reasonable amount of time. In addition, the images use one channel grey scale images in each depth layer. The ten frames from each data point used for the 3D image were not consecutive, but distributed equally throughout each video sequence.

For this experiment, neither of the base models used in previous experiments was applicable since they deal in two dimensions. As the construction of a completely novel model lies outside the scope of this paper, the model used for this experiment is given in [1], [7].

The results of this experiment shows that surprisingly, the 3D CNN with 32x32x10 input data resulted in the best accuracy score of all experiments so far, reaching a final accuracy of 94.8% after 100 epochs of training. Figure 3 shows the progression of the accuracy over the course of training and reveals that the trend of the accuracy score shows a slight upwards trend even in the end of training, suggesting that the

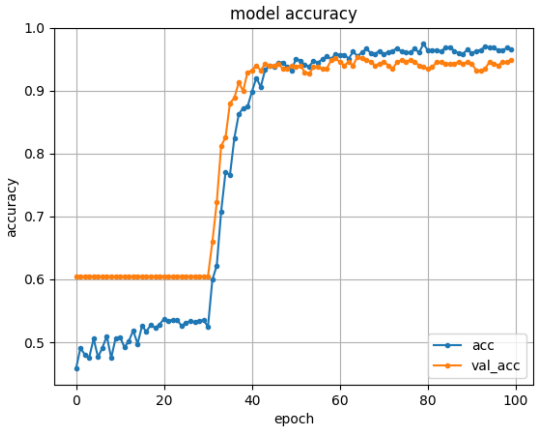

Fig. 3. The progression of accuracy and loss over 100 epochs of training on the 3DCNN model.

results could be improved even further by allowing it more epochs. At the same time, it reveals that the training accuracy is slightly higher than the validation accuracy in the end of the training phase, which possibly could indicate that the model is somewhat over-fitting.

D. Bidirectional LSTM and Hi vs. Low Dimension Input

At this stage, the 3D CNN has outperformed all other models for the relabeled AMI Corpus Meeting dataset used in these controlled experiments. However, the second best model is not far behind. The RCNN combining VGG16 features with LSTM cells reached 88.0% accuracy. Before settling on a model for further exploration, more experimentation is carried out on the RCNN model to determine whether we can improve performance.

The experiments in this section will explore two parameters: high vs. low dimension input, and unidirectional vs. bidirectional LSTM layers. The first aspect of dimensions refers to the fact that in the RCNN experiment, the three final fully connected layers of the original VGG16 model were retained before using the resulting features as input to the new RNN. These three layers are each scaling down the data, making the resulting input to the RNN lower dimensional. By removing these three layers, the RNN model on top would have higher dimensional data to work with, which could possibly help the performance.

The second aspect being tested here is the concept of bidirectional LSTMs. Normal unidirectional LSTMs retain a history of past processing, and thus achieving temporal processing. By implementing two LSTM layers next to each other of opposite direction, each data point also gets a reference to its future state as opposed to only its past. By allowing two time directions, information about past, current and future data can be processed at once [15].

In Table II we report results from these experiments. As expected, the unidirectional LSTM with low dimension input performed almost precisely the same as the last, most successful experiment done in section IV-B. This is of course a result of the fact that, that experiment used an identical model with VGG16 as a base model with pre initialized weights, unidirectional LSTM layer and low dimension input. Interestingly, using unidirectional LSTM with high dimension input resulted in an increase in accuracy, achieving 92%, almost tying the 3DCNN. Bidirectional LSTM layers however seem to not provide any improvement for either low or high dimensional input.

TABLE II
SCORES FROM DIRECTION/DIMENSION EXPERIMENTS

Direction	Dimension	Score
Uni	High	**92%**
Uni	Low	**88%**
Bi	High	**66%**
Bi	Low	**84%**

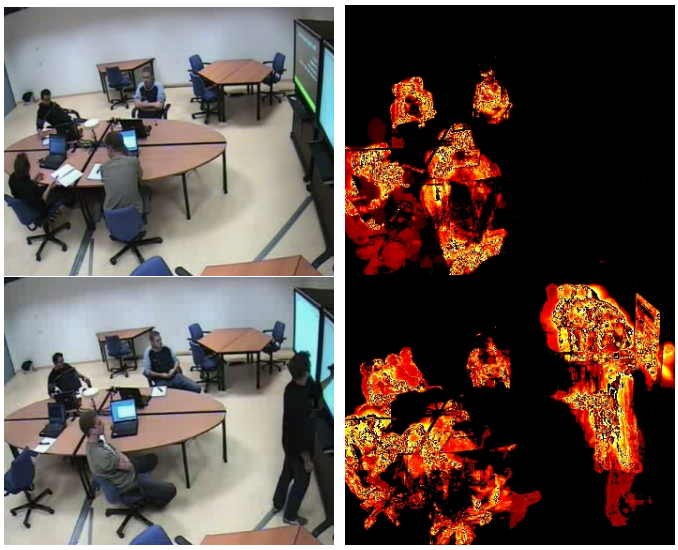

Fig. 4. Heat map representation of movements in a meeting and a presentation in the exact same environment and camera angle. Top: A meeting and its heatmap representation. Bottom: A presentation and its heat map representation.

E. Discussion

The experiments showed that the best performing model was the 3DCNN model. It is very likely that the key to understanding the success of the 3DCNN lies in the nature of the data and how it is classified, i.e. what high level features are the most determinant in deciding which class a frame sequence belongs to, and how well does the model extract those features. With regards to what the data looks like, each video sequence used for training and validation is filmed using a static camera indoors. This means that the vast majority of movement in each video segment is due to the people in the frame (if there are any). If a groups of pixels change from one frame to another, those pixels almost certainly represent a person in the frame. A very plausible explanation for why the 3DCNN performed so well is that the easiness of identifying people and moving objects gave it an advantage over the

VGG16 implementation. This is an inherent advantage of the combination of the network type and structure of the data.

The pre-trained VGG16 model is very effective in extracting high level features of images, and would almost certainly outperform the 3DCNN if determinant factors of classification included static objects in the video sequences. However, it is also possible that the general patterns of movement in the video sequences provided more information than expected.

In order to better understand how impactful these types of changes were between classes, we implemented a background subtraction algorithm, that essentially separates a moving foreground from a static background. The output for each frame is a binary matrix where each index represents whether the pixel is moving or not. These values were then used to create a heat map of movement, where brighter areas represent high movement activity throughout the video sequence.

As seen in Fig. 4, which portrays a meeting and a presentation from the same recording session and their respective movement heat maps. The type of movement appear to differ significantly between the two activities. In the meeting, movement is localized to the attendants seating positions, but in the presentation movement is much more spread out in the area where the presenting person is located. Another aspect seen in the heat map is that the screen present in the image has also generated changes in pixel values.

With that said, this analysis raises questions about the generalization and robustness of the 3DCNN technique in this problem space, especially when the test data set would contain video sequences that are far less similar in comparison with the training data set. So far, many, if not most, of the environments and precise camera angles are present in both the training and validation data. This fact has two sides to it. On the one hand, it is safe to say that the model has not over-fitted on specific features of frames, such as objects in view. On the other hand, it might have over-fitted the movement patterns of activities seen from very specific viewing angles. Therefore, the question yet to answer is whether the model has learned to abstract what type of movement makes a video sequence likely to belong to a certain class, or just learned to recognize the patterns of movement that are occurring in the precise environment of that data set. With his goal in mind, we design a experiment for a real-world setup, which we conduct in one of the university labs, in order to evaluate the extent to which the model can perform in a more dynamic setting.

V. Experiences in-the-wild : IoTaP dataset

During the controlled experiments the 3DCNN produced the best accuracy results for the AMI Corput Meeting data set, so this model was selected to be applied in real-world testing.

In order to capture real-world data of office activities, two cameras were mounted in the IOTAP lab of Malmö University. These cameras recorded for approximately one week during office hours and captured in total almost 160 hours of 640p video, or almost one terabyte of data. The data was then manually viewed, sorted and labeled to use in testing. Out of the two cameras, one was mounted in the corner of the lab in eye level, and one was mounted on the ceiling. Each video file is a sequence of between 2 minutes and 3 minutes 30 seconds.

Fig. 5. An image of a meeting captured from corner camera and its movement heat map .

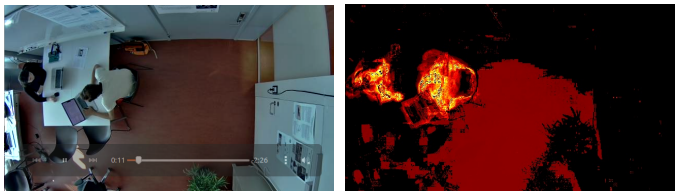

Fig. 6. An image of a meeting captured from ceiling camera, and its movement heat map

Two different strategies have been employed for evaluation. Namely, validating on randomly selected sequences, and validating on sequences separated by date of capture. According to the first strategy, 20% of the data will be selected randomly for the validation data set and the rest will be used for training. This gives insight into the model's accuracy in correctly classifying unseen video sequences that belong to (very likely) previously seen situations. The second strategy is to use the the first five days of captured data as training data, while the last days for validation. This is more challenging, but reflects on the model's capability of generalizing to completely unseen situations in seen environments.

For the first experiment, we use all data from all two cameras, randomly selecting 20% of the data for validation and then training the 3DCNN model for one hundred epochs, followed by validation. Note that the data is split, such that video sequences from just one camera at a time are used for both training and validation. The subset of data from the ceiling mounted camera reached a very high accuracy of 97%. Interestingly, the same experiment but with data only from the corner camera obtained a much worse results, of just 48.2%.

A very likely explanation for the poor results of the corner camera is that the activities shown are often much less spatially separated than in the case of the ceiling camera. As indicated by the heat map movement analysis made for the controlled experiments in section IV-E, the spatial separation of movements seems to be an important factor in determining the activity type. Figures 5 and 6 depict one example of a meeting with an associated heat map representation of the movement, for the corner and ceiling cameras respectively. Clearly, one can notice that the positioning and angle of the camera, relatively to where the activities are occurring has a very significant impact on the video scene recorded.

For the second experiment, a more challenging problem will be addressed. Specifically, whether the model can perform well on completely unseen data, depicting situations on different days than what was used in training. In this section, data from five of the days of video recording will be used for training, while another two days are used for validation. The data from the corner camera produced an accuracy of 45%, while the ceiling camera obtained 94.2%. Interestingly, the corner camera and ceiling camera only showed a small decline in accuracy when using the data from unseen dates, which is an important indication of the generalization capability of the model.

Conclusively, the hypothesis that the spatial separation of movement retrieved from the temporal depth inclusion of the 3DCNN is highly determinant for the models capacity of assessing activity class, lies very much in line with the analysis of the results found during the controlled experiments. Results from both experiments suggests that a camera filming office activities from an angle where people in the frame are spatially separated can have a high effectiveness of solving the task if combined with a 3DCNN model.

VI. Conclusions

This paper presents a comparative analysis in order to determine how the task of activity recognition in office scenarios could be solved using video data and deep learning. Two different data sets were used to investigate how to best construct a model to solve the task of classifying office activities. First, a subset of the AMI Meeting Corpus data set was manually re-labeled to fit the problem space, and served as a baseline for controlled experiments of different models. It was found that a pre-trained VGG16 model used to extract features as input to an RNN with a unidirectional LSTM layer performed very well on the data set, and the inclusion of temporal data was crucial in reaching high performance. It was also shown that a pre-trained model significantly outperformed a randomly initialized model, even though the model was pre-trained for object detection. The various experiment results also showed that the specific configuration of the RCNN was very important in finding the best model, and that performance varied greatly between different configurations.

The combined results from both the controlled experiments together with the real world testing strongly suggests that the task of activity recognition in office environments can be largely solved by using a 3DCNN and making sure that the camera angle is sufficient for spatial separation of the subjects movements. This paper provides an understanding of how different types of deep learning configurations perform on the given task and provides an example of a model that achieves above 94% accuracy under the right conditions.

References

[1] Fujimoto laboratory in kobe city college of technology. http://www.kobe-kosen.ac.jp/. [Online; accessed 23-May-2018].

[2] Fahed Alkhabbas, Majed Ayyad, Radu-Casian Mihailescu, and Paul Davidsson. A commitment-based approach to realize emergent configurations in the internet of things. In *IEEE International Conference on Software Architecture Workshops, ICSA Workshops*, pages 88–91, 2017.

[3] Wongun Choi, K. Shahid, and S. Savarese. What are they doing? : Collective activity classification using spatio-temporal relationship among people. In *2009 IEEE 12th International Conference on Computer Vision Workshops, ICCV Workshops*, pages 1282–1289, Sept 2009.

[4] J. Carletta et al. The ami meeting corpus: A pre-announcement. In Steve Renals and Samy Bengio, editors, *Machine Learning for Multimodal Interaction*, pages 28–39, Berlin, Heidelberg, 2006. Springer Berlin Heidelberg.

[5] M. S. Ibrahim, S. Muralidharan, Z. Deng, A. Vahdat, and G. Mori. A hierarchical deep temporal model for group activity recognition. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1971–1980, June 2016.

[6] A. Karpathy, G. Toderici, S. Shetty, T. Leung, R. Sukthankar, and L. Fei-Fei. Large-scale video classification with convolutional neural networks. In *2014 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1725–1732, June 2014.

[7] Fujimoto laboratory in Kobe City College of Technology. Fujimoto lab 3dcnn code repository. https://github.com/kcct-fujimotolab/3DCNN. [Online; accessed 23-May-2018].

[8] Radu-Casian Mihailescu and Paul Davidsson. Integration of smart home technologies for district heating control in pervasive smart grids. In *2017 IEEE International Conference on Pervasive Computing and Communications Workshops, PerCom Workshops 2017*, pages 515–520, 2017.

[9] Radu-Casian Mihailescu, Paul Davidsson, Ulrik Eklund, and Jan A. Persson. A survey and taxonomy on intelligent surveillance from a system perspective. *Knowledge Eng. Review*, 33:e4, 2018.

[10] Radu-Casian Mihailescu, Jan Persson, Paul Davidsson, and Ulrik Eklund. Towards collaborative sensing using dynamic intelligent virtual sensors. In Costin Badica, Amal El Fallah Seghrouchni, Aurélie Beynier, David Camacho, Cédric Herpson, Koen Hindriks, and Paulo Novais, editors, *Intelligent Distributed Computing*, pages 217–226. Springer International Publishing, 2017.

[11] Joseph Redmon, Santosh Kumar Divvala, Ross B. Girshick, and Ali Farhadi. You only look once: Unified, real-time object detection. *CoRR*, abs/1506.02640, 2015.

[12] Shaoqing Ren, Kaiming He, Ross B. Girshick, and Jian Sun. Faster R-CNN: towards real-time object detection with region proposal networks. *CoRR*, abs/1506.01497, 2015.

[13] Charissa Ann Ronao and Sung-Bae Cho. Human activity recognition with smartphone sensors using deep learning neural networks. *Expert Syst. Appl.*, 59(C):235–244, October 2016.

[14] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C. Berg, and Li Fei-Fei. Imagenet large scale visual recognition challenge. *International Journal of Computer Vision*, 115(3):211–252, Dec 2015.

[15] Mike Schuster and Kuldip K. Paliwal. Bidirectional recurrent neural networks. *Signal Processing, IEEE Transactions*, 1997.

[16] Pierre Sermanet, David Eigen, Xiang Zhang, Michaël Mathieu, Rob Fergus, and Yann LeCun. Overfeat: Integrated recognition, localization and detection using convolutional networks. *CoRR*, 2013.

[17] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *CoRR*, abs/1409.1556, 2014.

[18] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. Going deeper with convolutions. In *Computer Vision and Pattern Recognition (CVPR)*, 2015.

[19] Whitney and Intel. Counting people: Use opencv* for edge detection. https://software.intel.com/en-us/articles/opencv-at-the-edge-counting-people, Feb 2018. [Online; accessed 14-May-2018].

[20] Jian Bo Yang, Minh Nhut Nguyen, Phyo Phyo San, Xiao Li Li, and Shonali Krishnaswamy. Deep convolutional neural networks on multichannel time series for human activity recognition. In *Proceedings of the 24th International Conference on Artificial Intelligence*, IJCAI'15, pages 3995–4001. AAAI Press, 2015.

[21] Rayoung Yang and Mark W. Newman. Learning from a learning thermostat: Lessons for intelligent systems for the home. In *Proceedings of the 2013 ACM International Joint Conference on Pervasive and Ubiquitous Computing*, UbiComp '13, pages 93–102, New York, 2013. ACM.