

# A highly effective deep learning based escape route recognition module for autonomous robots in crisis and emergency situations

Ricardo Buettner and Hermann Baumgartl

Aalen University, Germany

ricardo.buettner@hs-aalen.de, hermann.baumgartl@hs-aalen.de

**Abstract**—Using convolutional neural networks we extend the work by Dugdale’s group on socially relevant multi-agent systems in crisis and emergency situations by giving the artificial agent the ability to precisely recognize escape signs, doors and stairs for escape route planning. We build an efficient recognition module consisting of three blocks of a depth-wise separable convolutional layer, a max-pooling layer, and a batch-normalization layer before dense, dropout and classifying the image. A rigorous evaluation based on the MCIndoor20000 dataset shows excellent performance values (e.g. over 99.81 percent accuracy). In addition, our module architecture is 78 times smaller than the MCIndoor20000 benchmark – making it suitable for embedding in operational drones and robots.

## 1. Introduction

While socially relevant multi-agent systems for crisis and emergency management received widespread attention from various research domains including computer science [1]–[3], civil engineering [4,5], electrical engineering [6], or aerospace engineering [7], a missing piece of applicable software agents is related to its ability to recognize situations and conclude action plans. An important step towards applicable agents was carried out by Bratman [8,9] who proposed the Belief-Desire-Intention (BDI) architecture [1,3]. BDI agents provide artificial agents with human-like decision making strategies, based upon their own sets of beliefs about their environments [10]–[12]. In this view an artificial agent needs to be aware of its environment [13].

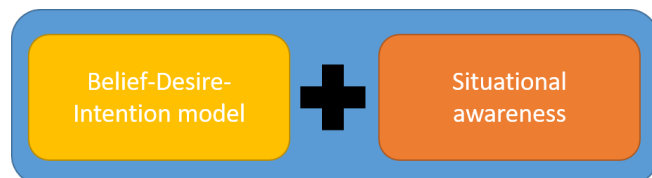


Figure 1: Dugdale et al. combined agent architecture

Another important step was taken by Dugdale’s group which proposed combining the BDI architecture and the Situational Awareness [14] theory by Endsley (Fig. 1). Endsley’s theory of Situational Awareness [14] explains human decision making in complex environments. According to this theory, “[s]ituation awareness is the perception of the elements in the environment within a volume of time and space,

the comprehension of their meaning, and the projection of their status in the near future.” (p.36). The combination of BDI architecture and Situational Awareness theory by Dugdale’s group leads to better agent decisions [15,16]. Research has shown that the implementation of sophisticated behavioral models (BDI, Situational Awareness theory, etc.) in software agents is necessary for real-world multi-agent systems [17]–[20].

The important missing piece for applicable software agents comprises the recognition of the environment as a precondition for understanding the current situation and to plan sophisticated actions. More precisely, the missing piece is related to level 1 of the Situational Awareness theory [14] – which consists of three consecutive levels:

### Level 1 Perception of elements in current situation:

All relevant elements and conditions of the environment are recognized – building the basis for all further levels of situational awareness.

### Level 2 Comprehension of current situation:

The elements which have been perceived in level 1 are now processed to understand the current situation in the context of the agents’ goals.

### Level 3 Projection of future status:

Based on the levels 1 and 2 possible future action of the elements are projected (action planning).

A significant practical weakness of most software agents is that they lack level 1 capabilities, so that they are not aware or only partially aware of their environment. However, these level 1 capabilities are necessary for sophisticated artificial agents’ decision making [13,16].

That is why by following the work by Dugdale’s group on socially relevant multi-agent systems we propose a high-class recognition module which substantially enhances level 1 capabilities. The module is based on convolutional neural networks [21]. Classical machine learning approaches for image recognition consist of two separate steps. In the first step, the so-called feature engineering stage, raw data is manually reduced to relevant representations. In the second step, the so-called classification stage, an algorithm tries to learn a mapping between these a priori generated features and an output variable (a target). The idea behind the convolutional neural networks is to combine these two steps into one, where the classification automatically shapes the feature engineering to extract more meaningful features with respect

to the classification results. Therefore these networks are capable of transforming raw data like pixel data from images into more meaningful forms of representation, so-called feature maps, to support the final decision on classification [21]. These feature maps represent characteristic regions of images, such as a nose in facial detection. Our recognition module makes use of the most current deep learning developments (e.g. convolutional neural networks [21], inception modules [22,23]), depth-wise-separable convolutions [24]). Since one of the most challenging tasks of level 1 perception is the visual recognition of objects and recognizing objects is also one of the most difficult problems in computer vision [24], we focus on object recognition in order to substantially contribute to making multi-agent systems practically much more relevant. We aim to develop a recognition module with high performance which is suitable for embedding into operational robots and drones.

The most important contributions are:

- 1) We build a highly effective recognition module with an accuracy of over 99.81 percent which significantly outperforms the current MCIndoor20000 benchmark [25].
- 2) Our module architecture is 78 times smaller than the MCIndoor20000 benchmark model and needs a significantly smaller amount of computational power [25], making it suitable for embedding in artificial drones and robots [26].
- 3) We extend the work by Dugdale's group on socially relevant multi-agent systems in crisis and emergency situations by giving the artificial agent the ability to precisely recognize objects [16].
- 4) We enhance the level 1 capabilities of Endsley's Situational Awareness theory [14].
- 5) Our approach closes the gap in visual level 1 Situational Awareness capabilities for multi-agent systems, showing the potential of artificial intelligence in crisis and emergency management [16].

The paper is organized as follows: Next we present an overview of related work before providing the research methodology, including the applied deep learning methods and the data used for evaluation. After that we show the machine learning results concerning the performance evaluation and analysis of the architecture. We then discuss the results and include theoretical and practical implications, before concluding with limitations and suggestions for future research.

## 2. Related Work

Within the last two years object recognition has substantially improved due to the very recent advancements in artificial neural networks, i.e. convolutional neural networks, transfer learning, VGG16, AlexNet, GoogLeNet, cf. table 1.

In particular, Ding et al. [27] proposed a deep convolutional neural network architecture with data augmentation for object recognition in synthetic aperture radar images (russian tanks, crashed planes, etc.). Their object recognition classifier achieved accuracies between 82.40 to 93.16

percent. Lagerstrom et al. [28] proposed using convolutional neural networks for feature extraction and Random Forests for classification to detect bushfires on CCTV images (86.60 percent accuracy). Arnold and Yamazaki [29] used convolutional neural networks (three layers) for their object classifier. Based on an own collected dataset from collapsed buildings they achieved accuracies from 76.17 to 83.64 percent. Bashiri et al. [25] – who collected the MCIndoor20000 dataset which we also use in our work here – applied transfer learning (AlexNet) and achieved 90.40 percent accuracy on the original non-redundant part of their dataset. Giannakeris et al. [30] presented a detection approach for classifying objects (e.g. flood, fire) in disaster scenarios. They used transfer learning based on VGG16 and achieved a very good accuracy of 97.5 percent. Muhammad et al. [31,32] used transfer learning based on AlexNet [31] and GoogLeNet (Inception-V3) [32] for fire detection on CCTV images – which raised accuracies to 94.39 and 94.43 percent.

However, our object recognition module will scrape on the 100 percent line and will substantially outperform all pre-existing classifiers (99.81 versus 82.40 to 97.50 percent accuracies).

## 3. Methodology

In order to substantially contribute to artificial intelligence research for crisis and emergency management and ensure strong methodological rigor, we followed the specific machine learning guidelines [21]–[24] and conducted a comprehensive literature review [33]–[35].

### 3.1. Convolutional Neural Networks (ConvNets)

Convolutional Neural Networks (ConvNets) consists of an input and an output layer, as well as multiple hidden layers (Fig. 2). The hidden ConvNets layers typically consist of convolutional layers, pooling layers, fully connected layers and normalization layers [21].

In recent years convolutional neural networks (ConvNets) have shown huge advancements in computer vision, establishing ConvNets as one of the prime algorithms for machine learning in computer vision [21]. Competing in the ImageNet competition – one of the most interesting computer vision challenges – ConvNets have shown remarkable performance enhancement in object recognition, segmentation and detection – outmatching human performance in object detection [22]. Mainly driven by the yearly ImageNet competition, ConvNets evolved from being deep stacks of classical convolutional layers (AlexNet, VGG) [36,37] to much more sophisticated architectures such as ResNet [38], Inception-V3 [22] and Xception [24].

Most major ConvNet architectures need high amounts of training data typically above 100,000 images. However in most real-world scenarios acquiring 100,000 or more images is very time consuming and therefore not applicable. A way to avoid the usage of very large datasets is the application of a transfer learning approach. This approach uses a pre-trained ConvNet, to 1) extract useful features from the

Author	Year	Method	Accuracy [percent]	Reference
Ding et al.	2016	Deep ConvNets with data augmentation	82.40 to 93.16	[27]
Lagerstrom et al.	2016	ConvNet + Random forest	86.60	[28]
Arnold and Yamazaki	2017	Small ConvNet for embedded systems	76.17 to 83.64	[29]
Bashiri et al.	2018	Transfer learning (AlexNet)	90.40	[25]
Giannakeris et al.	2018	Transfer learning (VGG16)	97.50	[30]
Muhammad et al.	2018	Transfer learning (AlexNet)	94.39	[31]
Muhammad et al.	2018	Transfer learning (GoogLeNet)	94.43	[32]

Table 1: Machine learning based object recognition models for crisis and disaster management.

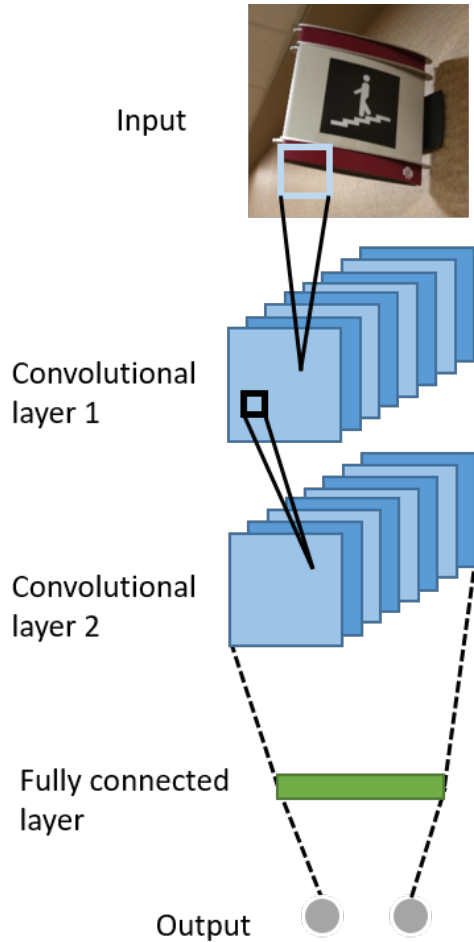


Figure 2: Example of a ConvNet architecture with 2 convolutional layer.

images or 2) extract features and fine-tune some of the weights to improve feature extraction. The transfer learning approach typically uses pre-trained ImageNet networks such as VGG or ResNet and their learned general representations of objects. The convolutional layers of the pre-trained networks provide features for fine tuning the final classifier based on the new domain-specific dataset [23,39].

### 3.2. General ConvNet architecture

Typically ConvNets consist of multiple convolutional layers. A convolutional layer for image data processes 3D arrays, with width, height and color channels. Images are processed by applying small filter-kernels on each image array, transforming the original input into feature maps. Common filter size include 3x3 and 5x5 patches, which are applied to every patch of the input data [21]. By applying the filters to every patch of the input data, spatial information is preserved. The result of every filter is passed through a non-linearty function like ReLU (rectified linear unit) and then passed to a pooling layer. These layers reduce the image size making the ConvNets more invariant to location. By combining multiple convolutional and pooling layers, ConvNets transform the image into more abstract feature representations [21,24].

While showing very good performance, current state-of-the-art ConvNets do require a reasonable amount of hardware resources. Especially classical architectures like AlexNet or VGG do have a higher amount of network weights, consuming more system memory and requiring more operations to compute. This makes them harder to implement into embedded system, especially into battery-powered devices like drones [23].

### 3.3. Inception modules

Classical ConvNet architectures for deep learning have a typical standard architecture consisting of multiple stacks of convolutional, max-pooling and normalization layers followed by fully-connected layers and dropout. Until the development of the inception module, ConvNets grew larger and larger, in order to learn more complex feature representations and gain greater classification performance [36,37]. The drawback of increasing the network size in depth and width is the heavy rise in network parameters and computational resources needed.

In normal convolutional layers the kernel is tasked with learning spatial feature representations and cross-channel representations at the same time. The inception module splits this combined operation into separate operations, where cross-channel correlations and spatial correlations are looked at independently. A standard inception module first covers the cross-channel correlations via a 1x1 convolution, mapping the input space into a smaller space,

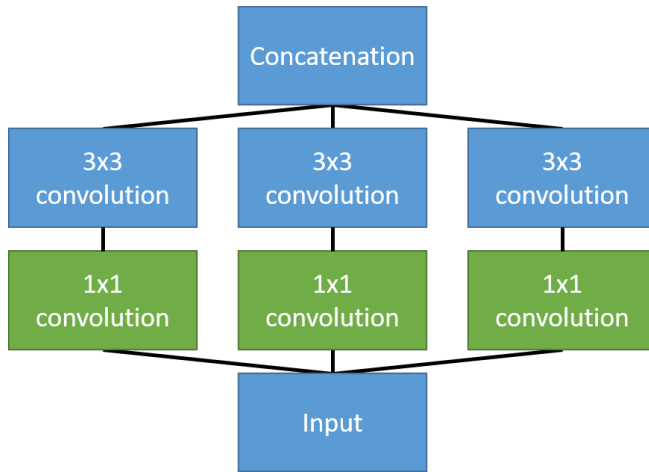


Figure 3: Simplified version of inception module from [24].

before handling spatial-correlations with a standard 3x3 or 5x5 convolution [22]. A simplified version of an inception module can be seen in Fig. 3. Inception modules have shown a great improvement in classification performance, while also decreasing the amount of model parameters [22,23].

The depth-wise separable convolution used in our architecture is based on inception modules. The depth-wise separable convolution implemented in many popular deep learning frameworks such as TensorFlow performs a spatial convolution over each input channel before doing a point-wise convolution to build a new channel space. So in the depth-wise separable convolution the order of operations is inverted from inception modules [24].

### 3.4. Evaluation data and data pre-processing

We evaluated our recognition module on the MCIndoor20000 dataset [25] consisting of 20,000 digital RGB images of hospital doors, signs and stairs. The dataset consists of two parts: the original dataset including 2,000 images and an extended dataset. The extended dataset consists of different variations of the original dataset modified by random rotations and four different types of artificial image noise. However these variations are copies of the original dataset, not containing any new images. Examples of each class are given in Fig. 4.

To exclude redundant images in training and avert a potential data leakage from training to testing data, we only used the original non-redundant and non-permuted images [40]. The total dataset size was 2,055 images. The images were resized to 224x224 pixels before [0,1]-normalizing. To rigorously evaluate the performance of the predictor we carried out a *hold-out cross validation*, a widely used performance evaluation approach. Hold-out cross validation splits the dataset into two separate parts: a training set solely used for training the model and an evaluation set which must not be shown to the model during training. The evaluation is only done on the evaluation data. Since the data is new to the



Figure 4: Example of each image class from [25].

model, potential overfitting of the model could be identified [40]. We used 1,541 images for training (75 percent) and 514 images for evaluation (25 percent).

## 4. Results

For training of the convolutional neural network we used Keras 2.1.5 package [41] with TensorFlow 1.8 as backend [42], running on a Nvidia GeForce GTX 1080 Ti. The training ran in a k-fold cross validation style on the training data, with 10 iterations [40]. Each iteration had 20 training epochs. The evaluation data has not been shown to the model during training.

### 4.1. Our recognition module architecture

The model we used to distinguish between the MCIndoor20000 classes consists of three blocks of depth-wise separable convolutions combined with a max-pooling layer and a batch-normalization layer. Our final model has three depth-wise separable convolutional layers, three max-pooling layers, three batch-normalization layers and two fully connected layers. The model received RGB colored images of shape 224x224x3. As shown in Fig. 5 the model transforms the images into high-level abstractions of size 7x7x32. In particular, the depth-wise separable convolutions help to reduce the amount of channels after each convolutional layer, making the model more light-weight



and computationally less expensive and thus applicable in artificial drones and robots [26].

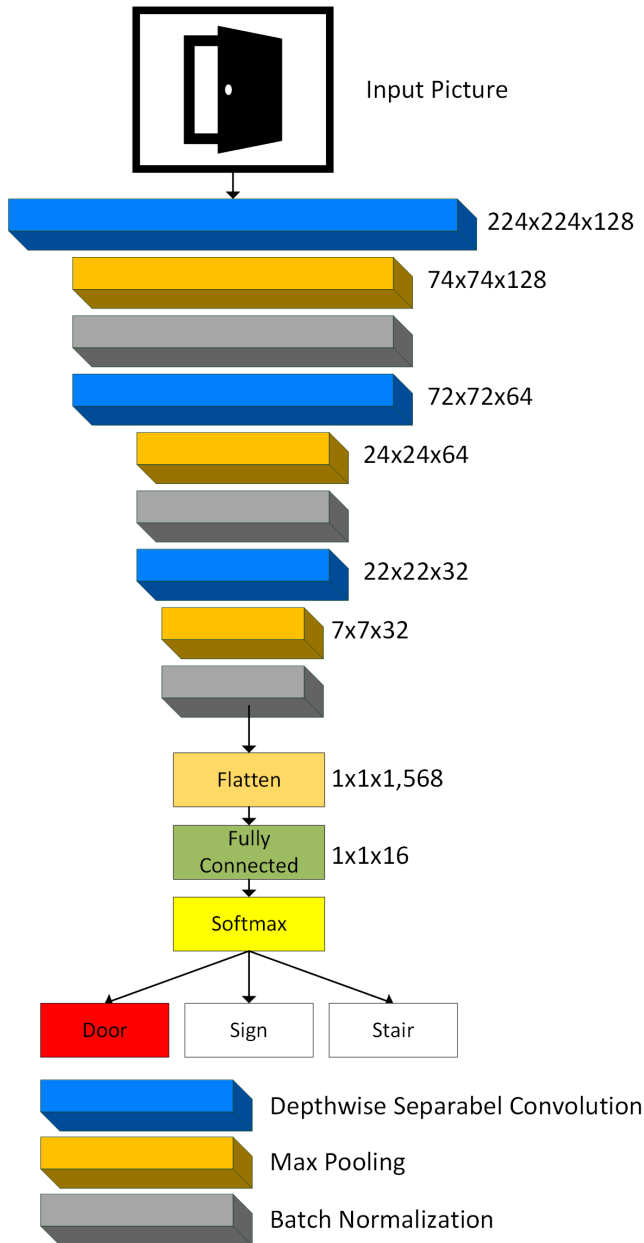


Figure 5: Our recognition module architecture.

#### 4.2. Performance evaluation

We evaluated the recognition module in terms of class-averaged sensitivity (true positive rate), precision (positive predictive value), Cohen’s Kappa score and accuracy. As shown in table 2 the classifier achieved excellent performance values.

The results show that the proposed convolution neural network has an accuracy of 99.805 percent in identifying doors, signs or stairs. Cohen’s-Kappa score is 99.707

Performance indicator	Value [percent]
Accuracy	99.805
True positive rate	99.805
True negative rate	99.899
Positive predictive value	99.807
Negative predictive value	99.701
Prevalence	33.334
Balanced accuracy	99.818
Kappa	99.707

Table 2: Evaluation indicators of our object recognition module.

percent. Prevalence is 33.334 percent. Positive predictive value is at 99.807 percent while negative predictive value is at 99.701 percent, underpinning the very good results and performance of our recognition module. Table 3 shows the confusion matrix of the evaluation dataset.

		Predicted		
		Door	Sign	Stair
Actual	Door	182	1	0
	Sign	0	179	0
	Stair	0	0	152

Table 3: Confusion matrix of our classifier based on the original image test dataset

As shown in table 3 only one object (i.e. a door) was misclassified as a sign. The classification of all 514 images took 2 seconds in total. The misclassified door is shown in Fig. 6. While the door image was falsely classified as a sign, the image also contains a small door sign which explains the misclassification.

To calculate the difference in size between AlexNet and our ConvNet architecture, we divided the amount of weights in the convolutional layers of AlexNet with the amount of weights in our architecture. According to Han et al. [43], AlexNet has a total of 62M parameters whereof 59M parameters belong to non-convolutional layers, while the remaining 3M weights reside in the convolutional layers. In comparison to AlexNet, our ConvNet architecture is 78 times smaller.

#### 5. Discussion

As demonstrated in table 2 our recognition module performs very well and achieves a very good classification performance. In addition, with an accuracy of 99.805 percent and a Kappa of 99.707 percent using the original dataset our module significantly outperforms the current benchmark by Bashiri et al. [25] by 9.4 percent. In addition, our recognition module outperforms all existing classifiers in terms of classification accuracy (cf. table 1). Put simply, our recognition module correctly classified every stair, every sign, and every door – with only a single exception. As shown in table 3 one



Figure 6: Sole misclassification within the evaluation set. While the door was falsely classified as a sign, the image also contains a small door sign.

door was misclassified as a sign. However, this misclassified door actually also contains a sign as shown in Fig. 6.

Since the benchmark by Bashiri et al. [25] also includes a prediction on the extended dataset including redundant images, we also tested our model on the extended dataset. This additional evaluation yielded very good results of 99.280 percent accuracy, also showing the model's resilience against artificial image noise and rotation.

### 5.1. Theoretical implications

Our recognition module allows the study of Endsley's situational awareness theory [14] in more detail. To empiri-

cally evaluate hypotheses from Endsley's theory, information systems scholars can systematically vary Endsley's proposed design features to study the effects on mental workload, stress and situational awareness.

In addition, we explicitly enhance the level 1 capabilities of Endsley's situational awareness theory. Our approach closes the gap in visual level 1 situational awareness capabilities in multi-agent systems, showing the potential of artificial intelligence in crisis and emergency management.

Finally, we extended the theoretical work by Dugdale's group on combining BDI-architecture and situational awareness by using the proposed recognition module.

### 5.2. Practical implications

We built an high-performance visual recognition module with an accuracy of over 99.81 percent which can be implemented in socially relevant software agents. The implementation of such level 1 situational awareness capabilities in software agents is an important precondition for further improvements on operable multi-agents systems as well as agent-based modeling and analyses in human decision making [13,16]. In addition, multi-agent based simulations for crisis disaster management can be coupled with our recognition module, providing a more comprehensive and holistic way of information presentation to crisis managers. Such an implementation most likely reduces mental workload, stress and human mistakes [14].

Since the recognition module is highly-efficient (78 times smaller than the MCIndoor20000 benchmark model), it can be implemented into robots or drones, providing a safe means of automatic information acquisition in dangerous emergency scenarios.

## 6. Conclusion

We built an efficient recognition module for object detection in emergency situations, to augment the visual recognition abilities of socially relevant multi-agent systems. Based on the most current deep learning developments (convolutional neural networks [21], inception modules [22,23]), depth-wise-separable convolutions [24]), we built a very effective and light-weight recognition module. The performance evaluation based on the MCIndoor20000 dataset showed that our model outperforms the benchmark by Bashiri et al. [25] in terms of all performance indicators and network size. As demonstrated in table 2 our model is capable of recognizing objects with a very good accuracy of 99.805 percent, only making one misclassification. Also the model shows good adoption to image noise. We found an architecture based on depth-wise separable convolutional layers, max-pooling layers and batch-normalization layers for efficient object recognition, outperforming the benchmark transfer learning approach [25]. In addition our network size is 78 times smaller than the benchmark model, making it also suitable for embedding into artificial drones and robots [26].

The recognition module enhances the level 1 capabilities which is necessary for sophisticated artificial agents' decision making [13,15,16].

### 6.1. Limitations

While the k-fold cross validation and the hold-out cross validation provide high internal validity, the external validity of our recognition module is not yet tested. Re-tests using another dataset as well as the implementation and evaluation in multi-agent systems is necessary to evaluate external validity. Another limitation is related to the current state of the MCIndoor20000 dataset. It provides labels for three classes: doors, signs and stairs but does not provide information about the subtypes of each class, like elevator doors or emergency staircases. Therefore the model cannot distinguish between subclasses of each object, which would be necessary in a real-world emergency scenario. Furthermore the model's resilience against image noise could only be tested using the artificial noise models provided by Bashiri et al. [25]. In real life emergency scenarios images can show additional noise coming from smoke, dust, fire, motion blur or bad lighting [29].

### 6.2. Future work

In future work **a)** we will show re-evaluation results of our recognition module based on another very large dataset which increases external validity and **b)** we will present the results of the implementation of our recognition module into an artificial drone which demonstrates the high performance in a real-world scenario (Fig. 7).

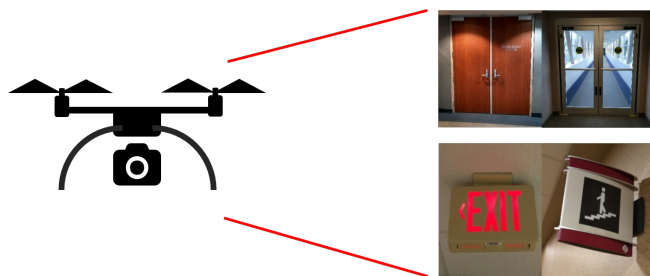


Figure 7: Implementation of recognition module in a drone.

In addition, we will extend our work by evaluating various agent coordination mechanisms (auctions [44]–[46] versus negotiation [47]–[50]) within real-world routing and escape scenarios [51].

Another future research line which we follow is related to the non-invasive evaluation of the reduction of a user's cognitive workload [52]–[56] during crisis and escape scenarios by autonomous escape agents.

Furthermore, to improve our recognition module, in future work we will include sub-class labelling in the MCIndoor20000 dataset. This enables a distinction to be made between sub-classes of doors, signs and stairs and to refine escape route recognition and subsequently escape route

planning. In addition, we improve our model architecture concerning the implementation of residual connections, cf. [38]. We will also benchmark our module to other existing approaches by runtime. Along with these refinements in data labeling we will also thoroughly test the model's resilience against other types of image noise such as dust or smoke. Beyond improvements of the model's performance and external validity, scholars could implement the recognition module into a multi-agent based simulation for emergency situations, such as that proposed by Mancheva et al. [16]. The operational capability of the recognition module could be further enhanced by scene recognition [57].

### Acknowledgments

We would like to thank the reviewers, who provided very helpful comments on the refinement of the HICSS paper. This research is partly funded by the German Federal Ministry of Education and Research (13FH4102IA).

### References

- [1] J. Bangate, J. Dugdale, E. Beck, and C. Adam, "SOLACE a multi-agent model of human behaviour driven by social attachment during seismic crisis," in *ICT-DM 2017 Proc.*, 2017, pp. 1–9.
- [2] D. Chen, L. Wang, A. Y. Zomaya, M. Dou, J. Chen, Z. Deng, and S. Hariri, "Parallel simulation of complex evacuation scenarios with adaptive agent models," *IEEE Transactions on Parallel and Distributed Systems*, vol. 26, no. 3, pp. 847–857, 2015.
- [3] F. Fiedrich and P. Burghardt, "Agent-based systems for disaster management," *Communications of the ACM*, vol. 50, no. 3, p. 41, 2007.
- [4] X. Pan, C. S. Han, K. Dauber, and K. H. Law, "Human and social behavior in computational modeling and analysis of egress," *Automation in Construction*, vol. 15, no. 4, pp. 448–461, 2006.
- [5] J. Shi, A. Ren, and C. Chen, "Agent-based evacuation model of large public buildings under fire conditions," *Automation in Construction*, vol. 18, no. 3, pp. 338–347, 2009.
- [6] E. Gelenbe and F.-J. Wu, "Large scale simulation for human evacuation and rescue," *Computers & Mathematics with Applications*, vol. 64, no. 12, pp. 3869–3880, 2012.
- [7] S. Sharma, H. Singh, and A. Prakash, "Multi-agent modeling and simulation of human behavior in aircraft evacuations," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 44, no. 4, pp. 1477–1488, 2008.
- [8] M. E. Bratman, *Intentions, Plan and Practical Reason*. Cambridge, Massachusetts: HUP, 1987.
- [9] M. E. Bratman, D. J. Israel, and M. E. Pollack, "Plans and Resource-Bounded Practical Reasoning," *Computational Intelligence*, vol. 4, no. 4, pp. 349–355, 1988.
- [10] J. Tweedale, N. Ichalkaranje, C. Sioutis, B. Jarvis, A. Consoli, and G. Phillips-Wren, "Innovations in multi-agent systems," *Journal of Network and Computer Applications*, vol. 30, no. 3, pp. 1089–1115, 2007.
- [11] A. S. Rao and M. P. Georgeff, "Bdi agents: From theory to practice," in *ICMAS-95 Proceedings*, 1995, pp. 312–319.
- [12] M. Wooldridge and N. R. Jennings, "Intelligent Agents: Theory and Practice," *Knowledge Engineering Review (KER)*, vol. 10, no. 2, pp. 115–152, 1995.
- [13] S. A. Mostafa, M. S. Ahmad, A. Ahmad, and M. Annamalai, "Formulating situation awareness for multi-agent systems," in *ACSAT 2013 Proceedings*. IEEE, 2013.
- [14] M. R. Endsley, "Toward a theory of situation awareness in dynamic systems," *Human Factors: The Journal of the Human Factors and Ergonomics Society*, vol. 37, no. 1, pp. 32–64, 1995.
- [15] L. Mancheva and J. Dugdale, "The design of an agent based model of human activities and communications in cardiac resuscitation," in *HICSS-48 Proc.*, 2015.

- [16] L. Mancheva and J. Dugdale, "Understanding communications in medical emergency situations," in *HICSS-49 Proc.*, 2016.
- [17] J. Bangate, J. Dugdale, C. Adam, and E. Beck, "A review on the influence of social attachment on human mobility during crises," in *ISCRAM 2017 Proc.*, T. Comes, F. Bénaben, C. Hanachi, and A. Montarnal, Eds., 2017, pp. 110–126.
- [18] C. Adam and B. Gaudou, "BDI agents in social simulations: a survey," *The Knowledge Engineering Review*, vol. 31, no. 03, pp. 207–238, 2016.
- [19] C. Adam, P. Taillandier, J. Dugdale, and B. Gaudou, "BDI vs FSM agents in social simulations for raising awareness in disasters," *International Journal of Information Systems for Crisis Response and Management*, vol. 9, no. 1, pp. 27–44, 2017.
- [20] G. Prati, E. Saccinto, L. Pietrantonio, and C. Pérez-Testor, "The 2012 northern Italy earthquakes: modelling human behaviour," *Natural Hazards*, vol. 69, no. 1, pp. 99–113, 2013.
- [21] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, no. 7553, pp. 436–444, 2015.
- [22] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions," in *CVPR 2015 Proc.*, 2015.
- [23] A. Canziani, E. Culurciello, and A. Paszke, "Evaluation of neural network architectures for embedded systems," in *2017 IEEE International Symposium on Circuits and Systems (ISCAS)*, 2017.
- [24] F. Chollet, "Xception: Deep learning with depthwise separable convolutions," in *CVPR 2017 Proc.*, 2017, pp. 1800–1807.
- [25] F. S. Bashiri, E. LaRose, P. Peissig, and A. P. Tafti, "MCIndoor20000: A fully-labeled image dataset to advance indoor objects detection," *Data in Brief*, vol. 17, pp. 71–75, 2018.
- [26] G. Hegde, Siddhartha, and N. Kapre, "CaffePresso: Accelerating Convolutional Networks on Embedded SoCs," *ACM Transactions on Embedded Computing Systems*, vol. 17, no. 1, pp. 1–26, 2018.
- [27] J. Ding, B. Chen, H. Liu, and M. Huang, "Convolutional neural network with data augmentation for SAR target recognition," *IEEE Geoscience and Remote Sensing Letters*, pp. 1–5, 2016.
- [28] R. Lagerstrom, Y. Arzhaeva, P. Szul, O. Obst, R. Power, B. Robinson, and T. Bednarz, "Image classification to support emergency situation awareness," *Frontiers in Robotics and AI*, vol. 3, 2016.
- [29] S. Arnold and K. Yamazaki, "Real-time scene parsing by means of a convolutional neural network for mobile robots in disaster scenarios," in *ICIA 2017 Proc.*, 2017.
- [30] P. Giannakeris, K. Aygerinakis, A. Karakostas, S. Vrochidis, and I. Kompatsiaris, "People and vehicles in danger: A fire and flood detection system in social media," in *IVMSP 2018 Proceedings*, 2018.
- [31] K. Muhammad, J. Ahmad, and S. W. Baik, "Early fire detection using convolutional neural networks during surveillance for effective disaster management," *Neurocomputing*, vol. 288, pp. 30–42, 2018.
- [32] K. Muhammad, J. Ahmad, I. Mehmood, S. Rho, and S. W. Baik, "Convolutional neural networks based fire detection in surveillance videos," *IEEE Access*, vol. 6, pp. 18 174–18 183, 2018.
- [33] J. Webster and R. T. Watson, "Analyzing the past to prepare for the future: Writing a literature review," *MIS Quarterly*, vol. 26, no. 2, pp. xiii–xxiii, 2002.
- [34] J. vom Brocke, A. Simons, B. Niehaves, K. Riemer, R. Plattfaut, and A. Clevén, "Reconstructing the Giant: On the Importance of Rigour in Documenting the Literature Search Process," in *ECIS '09 Proc.*, 2009, pp. 2206–2217.
- [35] J. vom Brocke, A. Simons, K. Riemer, B. Niehaves, R. Plattfaut, and A. Clevén, "Standing on the Shoulders of Giants: Challenges and Recommendations of Literature Search in Information Systems Research," *Communications of the Association for Information Systems*, vol. 37, 2015, article 9.
- [36] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in Neural Information Processing Systems 25*, 2012, pp. 1097–1105.
- [37] K. Simonyan, A. Vedaldi, and A. Zisserman, "Deep fisher networks for large-scale image classification," in *Advances in Neural Information Processing Systems 26*, 2013, pp. 163–171.
- [38] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *CVPR 2016 Proc.*, 2016.
- [39] H.-C. Shin, H. R. Roth, M. Gao, Le Lu, Z. Xu, I. Nogues, J. Yao, D. Mollura, and R. M. Summers, "Deep convolutional neural networks for computer-aided detection: Cnn architectures, dataset characteristics and transfer learning," *IEEE Transactions on Medical Imaging*, vol. 35, no. 5, pp. 1285–1298, 2016.
- [40] S. Arlot and A. Celisse, "A survey of cross-validation procedures for model selection," *Statistics Surveys*, vol. 4, pp. 40–79, 2010.
- [41] F. Chollet et al., "Keras," <https://keras.io>, 2015.
- [42] M. Abadi, P. Barham, J. Chen, Z. Chen, A. Davis, J. Dean, M. Devin, S. Ghemawat, G. Irving, M. Isard, M. Kudlur, J. Levenberg, R. Monga, S. Moore, D. G. Murray, B. Steiner, P. Tucker, V. Vasudevan, P. Warden, M. Wicke, Y. Yu, and X. Zheng, "Tensorflow: A system for large-scale machine learning," in *USENIX-OSDI 2016 Proc.*, 2016, pp. 265–283.
- [43] S. Han, H. Mao, and W. J. Dally, "Deep Compression: Compressing Deep Neural Network with Pruning, Trained Quantization and Huffman Coding," in *ICLR 2016 Proceedings*, 2016.
- [44] J. Landes and R. Buettner, "Job Allocation in a Temporary Employment Agency via Multi-dimensional Price VCG Auctions Using a Multi-agent System," in *In MICAI 2011: 10th Mexican International Conference on Artificial Intelligence, November 26 - December 4, 2011, Puebla, Mexico*. IEEE Computer Society Press, 2011, pp. 182–187.
- [45] R. Buettner and J. Landes, "Web Service-based Applications for Electronic Labor Markets: A Multi-dimensional Price VCG Auction with Individual Utilities," in *ICIW 2012 Proc.*, 2012, pp. 168–177.
- [46] R. Buettner, B. Daxenberger, and C. Woelsle, "User acceptance in different electronic negotiation systems - a comparative approach," in *In ICEBE 2013: Proceedings of the 10th IEEE International Conference on e-Business Engineering, September 11 - 13, Coventry, UK*. IEEE CS Press, 2013, pp. 1–8.
- [47] R. Buettner, "The State of the Art in Automated Negotiation Models of the Behavior and Information Perspective," *ITSSA*, vol. 1, no. 4, pp. 351–356, 2006.
- [48] R. Buettner, "A Classification Structure for Automated Negotiations," in *IEEE/WIC/ACM WI-IAT 2006 Proc.*, 2006, pp. 523–530.
- [49] R. Buettner and S. Kirm, "Bargaining Power in Electronic Negotiations: A Bilateral Negotiation Mechanism," in *EC-Web '08 Proceedings*, ser. LNCS, vol. 5183, 2008, pp. 92–101.
- [50] R. Buettner, "Cooperation in Hunting and Food-sharing: A Two-Player Bio-inspired Trust Model," in *BIONETICS '09 Proc., Avignon, France, December 9-11, 2009*, 2009, pp. 1–10.
- [51] J. Dallmeyer, R. Schumann, A. D. Lattner, and I. J. Timm, "Don't go with the Ant Flow: Ant-inspired Traffic Routing in Urban Environments," *Journal of Intelligent Transportation Systems: Technology, Planning and Operations*, vol. 19, no. 1, pp. 78–88, 2015.
- [52] R. Buettner, "Cognitive Workload of Humans Using Artificial Intelligence Systems: Towards Objective Measurement Applying Eye-Tracking Technology," in *KI 2013 Proc.*, ser. LNAI, vol. 8077, 2013, pp. 37–48.
- [53] R. Buettner, B. Daxenberger, A. Eckhardt, and C. Maier, "Cognitive Workload Induced by Information Systems: Introducing an Objective Way of Measuring based on Pupillary Diameter Responses," in *Pre-ICIS HCI/MIS 2013 Proc.*, 2013, paper 20.
- [54] R. Buettner, "Analyzing Mental Workload States on the Basis of the Pupillary Hippus," in *NeuroIS '14 Proc.*, 2014, p. 52.
- [55] R. Buettner, "Asking both the User's Brain and its Owner using Subjective and Objective Psychophysiological NeuroIS Instruments," in *ICIS 2017 Proceedings*, 2017.
- [56] R. Buettner, S. Sauer, C. Maier, and A. Eckhardt, "Real-time Prediction of User Performance based on Pupillary Assessment via Eye Tracking," *AIS Transactions on Human-Computer Interaction*, vol. 10, no. 1, pp. 26–56, 2018.
- [57] B. Zhou, A. Lapedriza, J. Xiao, A. Torralba, and A. Oliva, "Learning Deep Features for Scene Recognition using Places Database," in *NIPS'14 Proceedings*, 2014, pp. 487–495.