Visual Subterranean Junction Recognition for MAVs based on Convolutional Neural Networks

Sina Sharif Mansouri, Petros Karvelis, Christoforos Kanellakis, Anton Koval, and George Nikolakopoulos
Robotics Team
Department of Computer Science, Electrical and Space Engineering
Luleå University of Technology
Luleå SE-97187, Sweden

***Abstract*—This article proposes a novel visual framework for detecting tunnel crossings/junctions in underground mine areas towards the autonomous navigation of Micro Aerial Vehicles (MAVs). Usually mine environments have complex geometries, including multiple crossings with different tunnels that challenge the autonomous planning of aerial robots. Towards the envisioned scenario of autonomous or semi-autonomous deployment of MAVs with limited Line-of-Sight in subterranean environments, the proposed module acknowledges the existence of junctions by providing crucial information to the autonomy and planning layers of the aerial vehicle. The capability for a junction detection is necessary in the majority of mission scenarios, including unknown area exploration, known area inspection and robot homing missions. The proposed novel method has the ability to feed the image stream from the vehicles on-board forward facing camera in a Convolutional Neural Network (CNN) classification architecture, expressed in four categories: 1) left junction, 2) right junction, 3) left & right junction, and 4) no junction in the local vicinity of the vehicle. The core contribution stems for the incorporation of AlexNet in a transfer learning scheme for detecting multiple branches in a subterranean environment. The validity of the proposed method has been validated through multiple data-sets collected from real underground environments, demonstrating the performance and merits of the proposed module.**

***Index Terms*—Visual junction Detection, Convolutional Neural Network, Subterranean Autonomous Navigation, MAVs.**

I. Introduction

Miniature aerial robotics have shown increased robustness and technological performance in constrained and well defined lab environments. Nevertheless, a new era is emerging for this technology, envisioning the deployment in real-scale infrastructure environments and with the capability of demonstrating levels of high autonomy [1], [2]. A characteristic example of these developments is the integration of the Micro Aerial Vehicles (MAVs) in various underground mine inspection operations, which is also the final envisioned application scenario of this article and with an overall objective to target inspection of known and unknown areas with the aim to collect data for the asset owners that in the sequel can be further analyzed.

Subterranean environments are harsh, posing obstacles for flying vehicles, including too narrow/wide passages, reduced visibility due to rock falls, dust, wind gusts and lack of proper illumination, all of which constitute necessary the development of elaborated control, navigation, and perception modules for these aerial vehicles. This article proposes a novel method that identifies junctions that exist in front of the MAV, using the on-board forward facing visual sensor. More specifically, underground areas can have complex geometries with many crossings among the tunnels. These crossings have major impact in the navigation mission, since when not considered, they can lead to a crash on the tunnel surface or a wrong turn, thus decreasing the overall efficiency and performance of the aerial platforms towards a proper mission execution. Therefore, it becomes a critical navigation capability for the aerial vehicle to identify a junction and to provide this information to the aerial planner for enabling a more safe and optimal overall mission execution.

In this article, due to the limited amount of existing data-sets for different types of junctions in underground tunnels and general subterranean environments, the transfer learning approach [3] has been utilized as the training method. More specifically, the AlexNet [4] is selected for executing a transfer learning, mainly due to the success of AlexNet pre-trained Convolutional Neural Network (CNN) features and the promising results that have been obtained from several image classification data-sets with transfer learning on AlexNet [5]. Towards this approach, the junction images are extracted from the data-sets of: a) an underground mine in Chile [6], and b) underground tunnels in Sweden [7]. In the sequel, these data-sets are classified manually to four categories of: 1) *left junction*, 2) *right junction*, 3) *left & right junctions*, and 4) *no junctions* as depicted in Figure 1. Then, the last three layers of AlexNet are replaced to set new layers for the classification of the four categories of images and the network is trained from the junction data-sets. The obtained class for each image provides information from the local surroundings of the vehicle, which can later be utilized for the autonomous navigation in subterranean environments.

A. Related Works

Autonomous navigation in unknown environments requires environmental awareness, such as recognition of obstacles, junctions, dead-ends, etc. Moreover, navigation, based on vision based techniques for MAVs has received a significant attention the latest years and with a big variety of application

This work has been funded by the European Unions Horizon 2020 Research and Innovation Programme under the Grant Agreement No. 730302 SIMS.
Corresponding author's email: sinsha@ltu.se

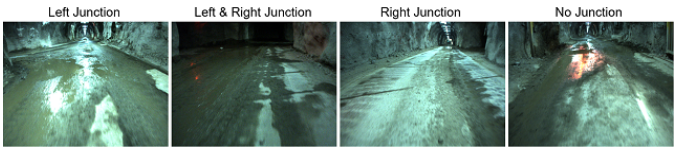

Fig. 1: Example of junctions' types in the training data-sets [6], [7].

scenarios [8], while it should be noted that the navigation, based on a forward looking camera, has been based mainly either on computer vision algorithms or on machine learning methods.

Towards computer vision based navigation, most of the works focused on obstacle detection methods, thus in [9], a mathematical model to estimate the obstacle distance to the MAV was implemented for collision avoidance. However, the method provided poor results at high velocities and low illumination environments. In [10] it was described the combination of multiple vision based components towards the navigation of autonomous aerial vehicles, while the proposed system used multiple sensing modalities for localization, mapping and obstacle free navigation. In this case, the obstacle avoidance scheme was consisted of 3 stereo cameras for a 360° coverage of the MAV's surroundings in the form of pointclouds. However, the proposed method relied on sufficient illumination, landmark extraction and high processing on-board power. In [11], random trees were generated to find the best branch, the method was evaluated in indoor environments and the paths were calculated on-line, while the occupancy map of the perceived environment was conducted. This method required in general a high computation power to process the images, to calculate the best next point and to accurately localize and store the previous information of the map in order to avoid revisiting the area. In general, the performance of the computer vision-based algorithms mainly relies on the surrounding environment with good distinctive features and good illumination and lighting conditions [9]. Furthermore, these methods require a high computation power to process the images and extract landmarks, factors that could limit the usage of these methods in real-life underground mine applications.

There are few works using machine learning techniques for the problem of navigation in in-door and out-door environments, mainly due to the fact that these methods require a large amount of data and a high computation power for training in most cases a CNN, which is an off-line procedure. However, after the training, the CNN can be used for enabling an autonomous navigation with much lower computation power, especially when compared to the training phase. The works using CNN for navigation, such as [12], [13], [14], utilized the image frame of on-board camera to feed the CNN for providing heading commands to the platform. These works have been evaluated and tuned in out-door environments and with a good illumination with the camera, thus providing rich data about the surrounding of the platforms, while none of the works consider the recognition of the junctions. In [15] a CNN binary classifier was proposed for outdoor road junction detection. Besides that, authors also considered the use of proposed architecture for navigation and experimentally evaluated it on commercially available MAV Bebop 2 from Parrot. The problem of junction detection for outdoor environments was also addressed by [16], where machine learning approach was used. In [17] authors propose an architecture that combines CNN, Bidirectional LSTM [18] and Siamese [19] style distance function learning for junction recognition in videos. In [18] a road intersection detection module has been proposed. The developed method addressed the problem as a binary classification, using Long-Term Recurrent Convolutional Network (LRCN) architecture to identify relative changes in outdoor features and eventually detecting intersections. These methods, mainly use binary classifier, while in real-life scenarios more complex type of junctions exist and the junction recognition should recognize the different types of junctions.

B. Contributions

Based on the aforementioned state-of-the-art the main contributions of this work are provided in this Section. Initially, the first and major contribution stems from the development of a vision based approach for junction detection, being among few works that study junction detection using monocular camera as the sensing modality. The system combines the AlexNet supervised CNN image classifier with transfer learning, introducing new classification categories. The proposed novel categories include 1) junction on the left side, 2) junction of the right side, 3) junctions on both the left and right side and 4) no junction, providing valuable information on the topological existence of junctions in the vehicle. The outcome is in the local surroundings of the vehicle leveraging the data stream from the single forward facing camera. The proposed method aims to have a general applicability for both high-end and lightweight aerial vehicles relying on single visual sensor, which can be found in both type of platforms. This is the reason why we used a wide known CNN [20] and Transfer Learning [21] making our method highly reproducible.

The second contribution of the proposed method stems from the evaluation process, using data-sets captured from real underground environments: 1) available online, and 2) from sites with limited access to the public, showing the applicability of the method in a variety of cases, enabling further developments in the field.

C. Outline

The rest of the article is structured as follows. Initially, Section II presents the AlexNet architecture and the corresponding transfer learning. Then, in Section III the data-set collection, the training of the network and the evaluation of the trained network is presented, while finally Section IV concludes the findings.

II. Utilizing AlexNet for Junction Recognition

This section initially provides a brief description of the AlexNet framework, while in the sequel the concept of transfer learning is explained.

A. AlexNet

AlexNet [4] is one of the most used and studied CNN methods [22] that has 60 million parameters and 650,000 neurons. Thus, a large data-set is required to train it, however due to the limited available data-sets of junctions, the transfer learning is selected in this article. In this approach, the input of the AlexNet is an Red, Green and Blue (RGB) image with fixed size of $227 \times 227 \times 3$ pixels and it follows with 2D convolutional layers of size 11×11 with an output size of $55 \times 55 \times 96$. Then it follows a 2D Max pooling layer of size 3×3 and an output of $27 \times 27 \times 96$, followed with 2D convolutional layers of 5×5 and an output of $27 \times 27 \times 256$. In the sequel there is another max pooling of size 3×3 and with an output of $13 \times 13 \times 256$, which passes through 2D convolutional layers of 3×3 with a same size output. Next, another 2D convolutional layers of 3×3 with an output size of $13 \times 13 \times 256$, followed with a max pooling of size 3×3 with an output of $6 \times 6 \times 256$. The output passes through two fully connected layers and the last layer results are fed into a softmax classifier with 1000 class labels. To summarize, AlexNet consists of eight layers, five of them that are convolutional layers and three of them that are fully connected layers. Each one of the first two convolutional layers are followed by an Overlapping Max Pooling layer. The other three convolutional layers (third, fourth and fifth) are connected directly. Finally the last convolutional layer (fifth) is followed by an Overlapping Max Pooling layer. Figure 2 depicts the overall utilized AlexNet structure.

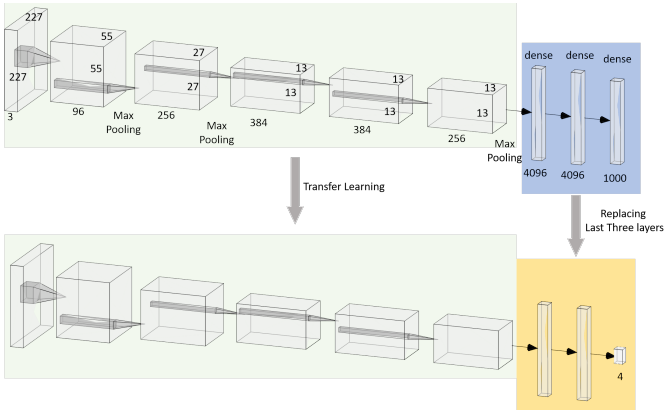

Fig. 2: AlexNet architecture and the transfer learning method.

Max pooling layers are usually used in CNNs in order to reduce the size of the matrices while keeping the depth the same. On the other hand overlapping max pooling uses adjacent windows which overlap each other in order to compute the max element from a window each time. It has been proven that this kind of max pooling reduces the top-1 and top-5 error rates [4].

Also, one of the main aspects of the AlexNet is the use of the Rectified Linear Unit (ReLU) [23]. The authors [4] proved that by using the ReLU nonlinearity, AlexNet could be trained a lot quicker than using classical activation functions like $sigmoid$ or $tanh$ [24]. Actually, they tested their hypothesis on the CIFAR-10 dataset [25] and the ReLU-AlexNet achieved the same performance (25% training error) with the Tanh-AlexNet in one sixth of the epochs.

B. Transfer Learning

Transfer learning [23], [26] for CNNs is usually referred as the process of using an already trained CNN in another data-set, where the number of classes to be recognized are different from the initial data-set, while it has been used in various problems and with various data-sets. There are two main strategies for Transfer Learnin, woth both of them to be using the same weights from the trained AlexNet on the images from the ImageNet database [27].

The first one treats the CNN as a feature extractor by removing the last fully connected layer. Then, one can use the features extracted from the trained AlexNet in order to train a classifier like [28] for the new data-set. The second one replaces the last connected layer and retrains the whole CNN for the new data-set. This allows for a fine tuning of the trained weights.

In this article, the last three fully connected layers of the AlexNet are replaced with a set of layers that will classify instead of 1000 classes the number of the desired 4 classes (no junction, left junction, right junction, left & right junction) as depicted in Figure 2.

III. Results

This section describes the data-sets from Chilean underground mine and Sweden underground tunnel, that the network was trained, while evaluation merits were also provided.

A. Data-Set

For transfer learning of the AlexNet, two data-sets from underground mine and tunnels are selected. The first one is from the Chilean underground mine data-set [6], which is collected by a Point Grey XB3 multi-baseline stereo camera, mounted with a forward facing orientation on the Husky A200. The camera was operated at $16\,\mathrm{fps}$ and with a resolution of 1280×960 pixels. The second data-set is collected from Luleå Sweden underground tunnels [7] which was collected manually with GoPro Hero 7 with resolution of 2704×1520 pixels and a frame rate of $60\mathrm{fps}$. In order to reduce the over-fitting of the CNN, the images from the cameras are down-sampled and few images are selected when the camera approaches to the junction and passes it. As an example, Figure 3 depicts the multiple images collected for the branch in the left of the tunnel. It should be highlighted that the data-sets from the Chilean underground mine contains more variety of branches, especially when compared to the Luleå Sweden underground tunnels data-set. Table I shows the overall number of images extracted from the video streams, due to the

Fig. 3: Examples of extracted images from the visual camera in case of a left branch in the tunnel. The images are extracted while the camera approaches to the junction and passes it (continuous with a direct heading).

Frame Per Second (fps) of the camera most of the images are similar in both data-sets and only 488, 339, 333, and 350 images are extracted for *left junction*, *left & right junctions*, *right junction*, and *no junction* respectively.

TABLE I: The number of extracted images for each category from the two data-sets, while the redundant images are excluded from the data-set.

	left	*left & right*	*right*	*no junction*
Chilean mine data-set	339	279	239	250
Sweden tunnels data-set	149	60	104	100

In the sequel, the data-set is manually classified to four categories of *left junction*, *right junction*, *left & right junctions*, and *no junction*. Figures 4 and 5 depict sample images of different areas of the Chilean underground mine data-set and the Luleå Sweden underground tunnels respectively.

Fig. 4: Examples of acquired images from the Chilean underground mine data-set [6].

B. Training and Evaluations of the CNN

Both data-sets are combined for training the AlexNet, while the junctions that are not included in the training data-set are

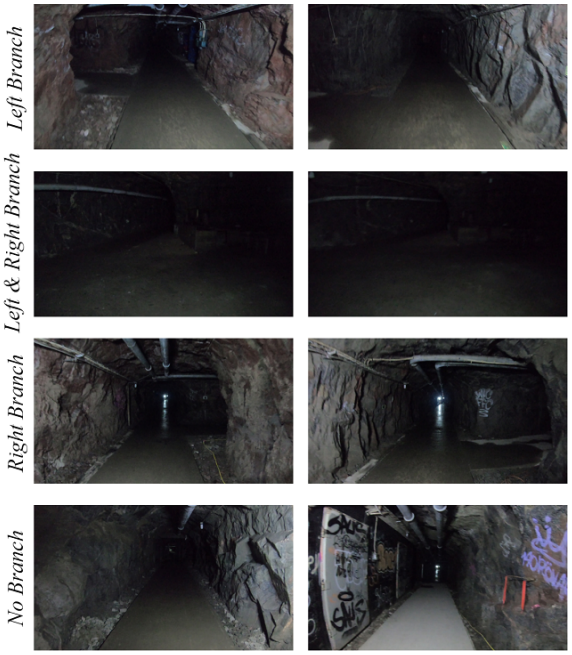

Fig. 5: Examples of acquired images from the Luleå Sweden underground tunnels.

used for validation of the network. Moreover, the images are resized to $227 \times 227 \times 3$ pixels. The network was trained on a workstation equipped with an Nvidia GTX 1070 GPU with mini-batch size of 10, maximum number epochs of 6, a selected initial learning rate of 10^{-4} and solved by the stochastic gradient descent [29] with momentum optimizer. The trained network provides an accuracy of 100% and 89.2% on training and validation data-sets respectively. Figure 6 shows the accuracy and loss of the training and validation data-set respectively, while the loss function for multi-class classification is defined as a cross entropy loss [23], [30].

Moreover, Figure 7 depicts the confusion matrix of the validation data-set, while the rows correspond to the predicted class from the validation data-sets and the columns correspond to the actual class of the data-set. The diagonal cells show the number and percentage of the correct classifications by the trained network. As an example, in the first diagonal, 45 images are correctly classified to the *left branch* category, which corresponds to 25.9% of the overall number of images. Similarly, 20 cases are correctly classified to *left & right*

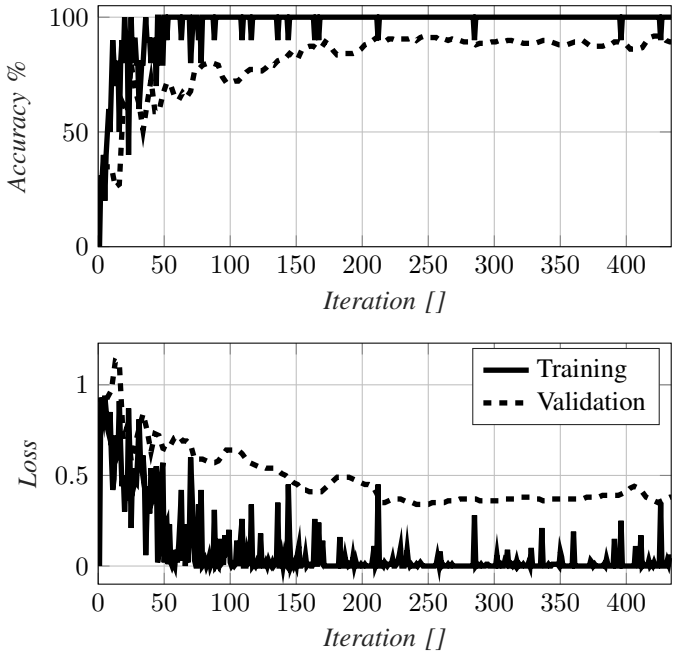

Fig. 6: Accuracy and loss of the network on training and validation data-sets.

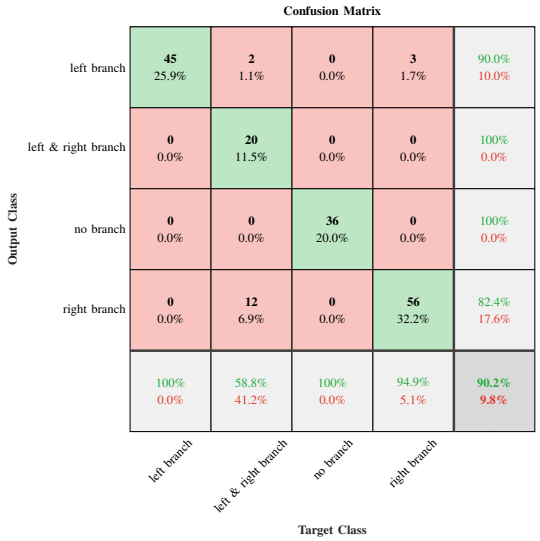

Fig. 7: The confusion matrix from the validation data-set.

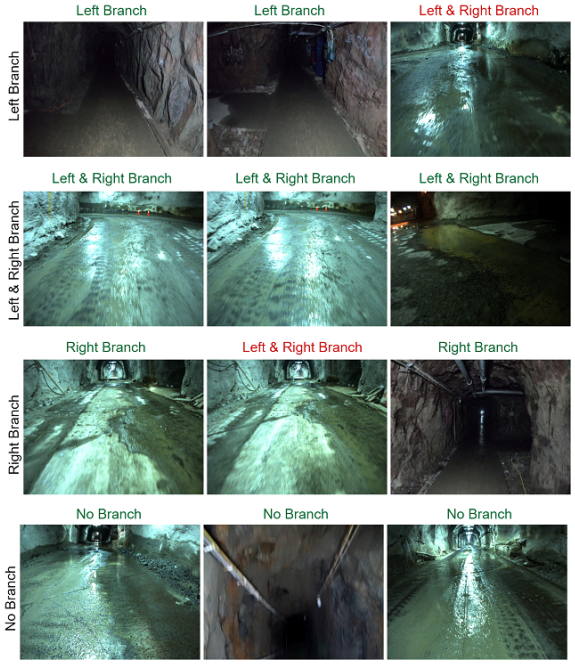

Fig. 8: Examples of validation data-set, while the correct label is written in the left of the images and the estimated class form AlexNet is written on top of each image.

branch that correspond to 19.5% of all the validation data-set. Moreover, the off-diagonal cells correspond to incorrectly classified observations, e.g. 2 and 3 images from the left branch are incorrectly classified to left & right branch and right branch respectively, which corresponds to 10% of the left branch validation data-set or 1.1% and 1.7% of all the data-sets respectively. Similarly, for the right branch 12 images are incorrectly classified to left & right branches. Furthermore, the right gray column displays the percentages of all the images predicted to belong to each class that are correctly and incorrectly classified. On the other hand, the bottom row depicts the percentages of all the examples belonging to each class that are correctly and incorrectly classified. The cell in the bottom right of the plot shows the overall accuracy. Overall, 90.2% of the predictions are correct and 9.8% are wrong.

Moreover, Figure 8 depicts 12 images from the validation data-set, while the correct label and the classification of the AleXNet are shown for each image. It should be highlighted that, these images are excluded from the training data-sets. It is observed that the network has incorrectly classified the *left junction* and *right junctions* images to *left & right junctions*, these images are from Chilean data-set.

Furthermore, Table II compares the training time, accuracy percentage and loss [23] between three well-known pre-trained CNN architectures AlexNet, GoogleNet [31], and the Inceptionv3Net [32]. As one can see from this table, the validation accuracy is smaller for the GoogleNet and significantly smaller for the Inception3Net. We think that this is due to the large number of Convolutional layers of these two Networks when compared to the AlexNet. Thus, more data-set is needed for these two networks and this support our choice for the AlexNet network.

IV. Conclusions

This work proposed a novel framework based on CNN for detecting tunnel crossing/junctions in underground mine areas, envisioning the application of MAVs autonomous deployment for inspection purposes. Within the emerging field of underground MAVs junction detection it has been iden-

TABLE II: The comparison of transfer learning between AlexNet, GoogleNet, and Inceptionv3Net.

	AlexNet	GoogleNet	Inceptionv3Net
Training Time [sec]	995	891	1438
Training Accuracy	100%	100%	100%
Validation Accuracy	89.2%	74.1%	63.79%
Training Loss	0.01	0.01	0.17
Validation Loss	0.29	0.82	0.93

tified as a fundamental capability for the aerial vehicle's autonomous navigation. Moreover, inspired by the concept of lightweight aerial vehicles, this work aims to provide a generic solution, keeping the hardware complexity low, relying only on a single visual sensor. It feeds the image stream from the vehicle's on-board forward facing camera in a CNN classification architecture expressed in four categories: 1) left junction, 2) right junction, 3) left-right junction, and 4) no junction in the local vicinity of the vehicle. The AlexNet model has been incorporated in a transfer learning scheme for the novel proposed classification categories, detecting multiple branches underground. The method has been validated by using data-sets collected from real underground environments, demonstrating it's performance and merits.

REFERENCES

[1] C. Kanellakis, E. Fresk, S. S. Mansouri, D. Kominiak, and G. Nikolakopoulos, "Autonomous visual inspection of large-scale infrastructures using aerial robots," *arXiv preprint arXiv:1901.05510*, 2019.

[2] C. Kanellakis, S. S. Mansouri, G. Georgoulas, and G. Nikolakopoulos, "Towards Autonomous Surveying of Underground Mine Using MAVs," in *International Conference on Robotics in Alpe-Adria Danube Region*. Springer, 2018, pp. 173–180.

[3] L. Torrey and J. Shavlik, "Transfer learning," in *Handbook of research on machine learning applications and trends: algorithms, methods, and techniques*. IGI Global, 2010, pp. 242–264.

[4] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in neural information processing systems*, 2012, pp. 1097–1105.

[5] M. Huh, P. Agrawal, and A. A. Efros, "What makes imagenet good for transfer learning?" *arXiv preprint arXiv:1608.08614*, 2016.

[6] K. Leung, D. Lühr, H. Houshiar, F. Inostroza, D. Borrmann, M. Adams, A. Nüchter, and J. Ruiz del Solar, "Chilean underground mine dataset," *The International Journal of Robotics Research*, vol. 36, no. 1, pp. 16–23, 2017.

[7] S. S. Mansouri, C. Kanellakis, G. Georgoulas, and G. Nikolakopoulos, "Towards MAV navigation in underground mine using deep learning," in *IEEE International Conference on Robotics and Biomimetics (ROBIO)*, 2018.

[8] C. Kanellakis and G. Nikolakopoulos, "Survey on computer vision for uavs: Current developments and trends," *Journal of Intelligent & Robotic Systems*, pp. 1–28, 2017.

[9] S. Saha, A. Natraj, and S. Waharte, "A real-time monocular vision-based frontal obstacle detection and avoidance for low cost uavs in gps denied environment," in *2014 IEEE International Conference on Aerospace Electronics and Remote Sensing Technology*. IEEE, 2014, pp. 189–195.

[10] F. Valenti, D. Giaquinto, L. Musto, A. Zinelli, M. Bertozzi, and A. Broggi, "Enabling computer vision-based autonomous navigation for unmanned aerial vehicles in cluttered gps-denied environments," *2018 21st International Conference on Intelligent Transportation Systems (ITSC)*, pp. 3886–3891, 2018.

[11] A. Bircher, M. Kamel, K. Alexis, H. Oleynikova, and R. Siegwart, "Receding horizon next-best-view planner for 3d exploration," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2016, pp. 1462–1468.

[12] S. P. Adhikari, C. Yang, K. Slot, and H. Kim, "Accurate natural trail detection using a combination of a deep neural network and dynamic programming," *Sensors*, vol. 18, no. 1, p. 178, 2018.

[13] L. Ran, Y. Zhang, Q. Zhang, and T. Yang, "Convolutional neural network-based robot navigation using uncalibrated spherical images," *Sensors*, vol. 17, no. 6, p. 1341, 2017.

[14] N. Smolyanskiy, A. Kamenev, J. Smith, and S. Birchfield, "Toward low-flying autonomous MAV trail navigation using deep neural networks for environmental awareness," *arXiv preprint arXiv:1705.02550*, 2017.

[15] S. Kumaar, S. Mannar, S. Omkar *et al.*, "Juncnet: A deep neural network for road junction disambiguation for autonomous vehicles," *arXiv preprint arXiv:1809.01011*, 2018.

[16] H. Haiwei, Q. Haizhong, X. Limin, and D. Peixiang, "Applying cnn classifier to road interchange classification," in *2018 26th International Conference on Geoinformatics*. IEEE, 2018, pp. 1–4.

[17] A. Kumar, G. Gupta, A. Sharma, and K. M. Krishna, "Towards view-invariant intersection recognition from videos using deep network ensembles," in *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2018, pp. 1053–1060.

[18] D. Bhatt, D. Sodhi, A. Pal, V. Balasubramanian, and M. Krishna, "Have i reached the intersection: A deep learning-based approach for intersection detection from monocular cameras," in *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2017, pp. 4495–4500.

[19] W.-t. Yih, K. Toutanova, J. C. Platt, and C. Meek, "Learning discriminative projections for text similarity measures," in *Proceedings of the fifteenth conference on computational natural language learning*. Association for Computational Linguistics, 2011, pp. 247–256.

[20] S.-H. Wang, S. Xie, X. Chen, D. S. Guttery, C. Tang, J. Sun, and Y.-D. Zhang, "Alcoholism identification based on an alexnet transfer learning model," *Frontiers in Psychiatry*, vol. 10, p. 205, 2019. [Online]. Available: https://www.frontiersin.org/article/10.3389/fpsyt.2019.00205

[21] M. Hussain, J. J. Bird, and D. R. Faria, "A study on cnn transfer learning for image classification," in *Advances in Computational Intelligence Systems*, A. Lotfi, H. Bouchachia, A. Gegov, C. Langensiepen, and M. McGinnity, Eds. Cham: Springer International Publishing, 2019, pp. 191–202.

[22] H. Shin, H. R. Roth, M. Gao, L. 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, May 2016.

[23] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. MIT Press, 2016, http://www.deeplearningbook.org.

[24] A. Saxena, "Convolutional neural networks (cnns): An illustrated explanation," *URL https://xrds. acm. org/blog/2016/06/convolutional-neural-networks-cnns-illustrated-explanation/. Last updated*, pp. 06–29, 2016.

[25] A. Krizhevsky, V. Nair, and G. Hinton, "Cifar-10 (canadian institute for advanced research)," 2010. [Online]. Available: http://www.cs.toronto.edu/ kriz/cifar.html

[26] S. J. Pan and Q. Yang, "A survey on transfer learning," *IEEE Transactions on Knowledge and Data Engineering*, vol. 22, no. 10, pp. 1345–1359, Oct 2010.

[27] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "ImageNet: A Large-Scale Hierarchical Image Database," in *CVPR09*, 2009.

[28] M. A. Hearst, "Support vector machines," *IEEE Intelligent Systems*, vol. 13, no. 4, pp. 18–28, Jul. 1998. [Online]. Available: http://dx.doi.org/10.1109/5254.708428

[29] K. P. Murphy, *Machine learning: a probabilistic perspective*. MIT press, 2012.

[30] P. Kim, "Matlab deep learning," in *With Machine Learning, Neural Networks and Artificial Intelligence*. Springer, 2017.

[31] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 1–9.

[32] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, "Rethinking the inception architecture for computer vision," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 2818–2826.