Towards Explainability of UAV-Based Convolutional Neural Networks for Object Classification

Chester Dolph,[1] Loc Tran[2], and B. Danette Allen[3]
NASA Langley Research Center, Hampton, VA, 23681

Establishing a basis for certification of autonomous systems using trust and trustworthiness is the focus of Autonomy Teaming and TRAjectories for Complex Trusted Operational Reliability (ATTRACTOR), a new NASA Convergent Aeronautical Solutions (CAS) Project. One critical research element of ATTRACTOR is *explainability* of the decision-making across relevant subsystems of an autonomous system. The ability to explain why an autonomous system makes a decision is needed to establish a basis of trustworthiness to safely complete a mission. Convolutional Neural Networks (CNNs) are popular visual object classifiers that have achieved high levels of classification performances without clear insight into the mechanisms of the internal layers and features. To explore the explainability of the internal components of CNNs, we reviewed three feature visualization methods in a layer-by-layer approach using aviation related images as inputs. Our approach to this is to analyze the key components of a classification event in order to generate component labels for features of the classified image at different layers of depths. For example, an airplane has wings, engines, and landing gear. These could possibly be identified somewhere in the hidden layers from the classification and these descriptive labels could be provided to a human or machine teammate while conducting a shared mission and to engender trust. Each descriptive feature may also be decomposed to a combination of primitives such as shapes and lines. We expect that knowing the combination of shapes and parts that create a classification will enable trust in the system and insight into creating better structures for the CNN.

I. Nomenclature

ATTRACTOR	=	Autonomy Teaming and TRAjectories for Complex Trusted Operational Reliability
CNN	=	Convolutional Neural Network
fps	=	Frames per Second
GPU	=	Graphics Processing Unit
R-CNN	=	Region-based Convolution Neural Network
SARUC	=	Search and Rescue Under the Canopy
UAV	=	Unmanned Aerial Vehicle

II. Introduction

Great interest exists in using deep learning methods to solve image classification problems [1]. One area of recent research interest is greater understanding of the internal mechanisms within neural networks. The process of how deep neural networks make decisions is not fully understood. The meaning of the internal components and why image features are chosen in the training process for a given layer is not clear. This concept of understanding the internal workings is called neural network explainability or interpretability. Deep network interpretability is a challenging problem because the internal components are non-linear representations of 2D images at varying levels of feature extraction with complex patterns that are visually unintuitive [1]. Feature visualization is a method of producing a visual representation of a network at the neuron, channel, or layer level. The visualization may be a manifestation of weights, convolution filters, activations, gradients, neurons, the response to a given input image, an amplification of

[1] Aerospace Engineer, Aeronautics Systems Engineering Branch, MS 238, AIAA Member
[2] Computer Engineer, Flight Software Systems Branch, MS 472
[3] NASA Senior Technologist (ST) for Intelligent Flight Systems, MS 233, AIAA Senior Member.

neural activity, reconstruction of the input from the response, or a combination of the aforementioned feature visualization techniques.

Currently, a CNN is often used as a black box where users train the network using an image dataset, supply an input image, and the output is a classification with little to no supporting evidence to why the classification is made. In this work, we explore context to classification through insights extracted from intermediate layers of a deep CNN. A deep CNN is composed of multiple layers between the input and output. Researchers have been tuning CNNs to get improved classification accuracy and speed for their application while forgoing explainability of the system. We are developing a system to extract intermediate information in a layer-by-layer approach. In addition to a classification, we will generate labels for information in the previous layer. We also present a survey of feature visualization techniques applied to aviation imagery and compare their outputs. The goal of this work is to achieve a greater understanding and explainability of CNN while focusing on aviation-related imagery. Reducing or even eliminating the "black box" implementation of image classification via *explainability* is critical to effective teaming of humans and machines where establishing trust between agents executing a shared mission. This is helpful in actual operations as well as testing and development.

III. Background

A. Deep Convolutional Neural Networks

Convolutional Neural Networks (CNN) have received substantial interest due to their high classification accuracies in image classification competitions (e.g., ImageNet [2]) and ease of training with automated learning for applications such as speech recognition, optical character recognition, and image based object recognition. Array-based multiplication is used extensively in the training of CNNs, thus substantial advances in the computational performance of GPUs in the last decade have enabled CNNs with over 100 layers and real-time systems on small UAVs using embedded computers. CNNs are composed of a sequence of layers. The first layers are typically convolution and max pooling. The convolution layer convolves the input of the layer by a filter from a filter bank. The filters are generated during the training or transfer learning process. The filters or kernels extract different type of information from an input (e.g. contours from an image). The output of the convolution layer is passed to a pooling layer where it is subsampled and passed to another convolution layer as an input. The final layer classifies the output of the previous layer using a probability function.

The architecture of deep neural networks utilize a series of convolution and pooling layers followed by several fully connected layers, resulting in a complex neural network where the internal weights of the neurons may not be readily understood by humans.

B. Region-based Convolutional Neural Networks (R-CNN)

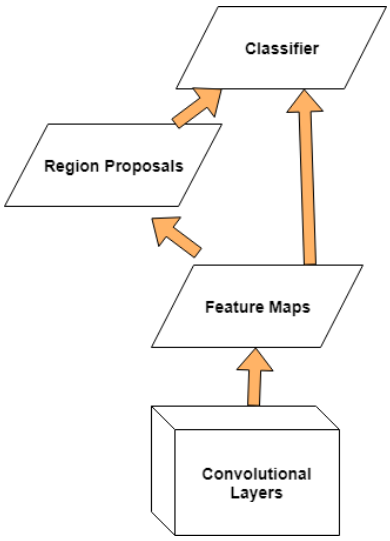


Fig. 1 Network structure of Faster R-CNN.

This section details Faster R-CNN, a recent CNN approach that efficiently combines object classification with an object detection module [3]. In a deep convolutional network, the first few layers are generally convolutional layers that encapsulate low level image primitives such as edges, corners, and spots. These primitives are the foundation to a variety of computer vision tasks. By reusing these low level layers, Faster R-CNN is capable of both detecting potential objects and classifying the objects while only computing the first convolutional layers once. **Fig. 1** shows the structure of Faster R-CNN where the region proposals are generated in one branch of the network. The result of the proposals is used to select the feature maps for the classification task.

Previous work completed by our group [4] includes implementing a custom trained network based on the Faster R-CNN framework for real-time obstacle detection onboard a small UAV. The algorithm was deployed on an NVIDIA Jetson TX2 and could achieve 3 fps on the embedded computer. **Fig. 2** shows a sample output from the network simultaneously detecting trees, persons, and drones indoors. The work showed that UAV-based embedded object classification was feasible especially as network structures continue to be optimized and hardware continues to advance. We were inspired to explore the explainability aspect of neural network decision making from this work to add trustworthiness to autonomous mission using CNNs. While the performance of the state of the art CNNs such as Faster-RCNN are impressive, CNNs are not transparent in nature. Determining what in the image causes the algorithm to choose one classification over another is difficult and more information is needed to build confidence in the resulting classification.

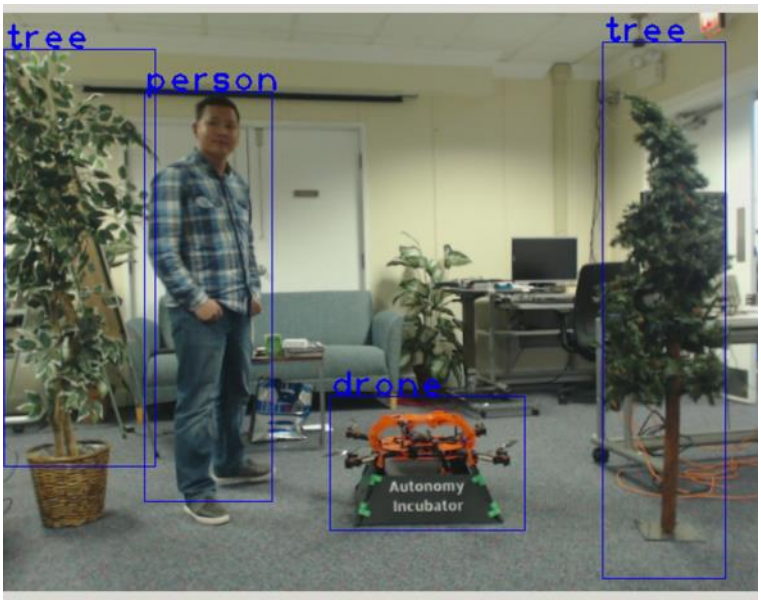


Fig. 2 Faster R-CNN output showing simultaneous detections of a person, trees, and a drone indoors.

C. Inception Network

In this paper, we focus on visualizing features from the Inception network [5] which is a popular CNN approach. The Inception network introduced an efficient method to combine multiple types of layers into one module. The network could then capture details at various scales such as with convolution kernels with different sizes. When customizing the structure of a CNN, a researcher must select the ordering, size, and shape of a layer. For example, a layer could be a 3x3 convolution, 5x5 convolution, a pooling layer, a fully connected layer, or many other types. In [5], the researchers chose to implement a combination of multiple types into one layer which they called an Inception module. An example of an Inception module is shown in Fig. 3 where three different convolutional layers are combined with a pooling layer. The researchers show that by performing 1x1 convolutional layers before the 3x3 and 5x5 convolutional layers, the dimensionality can be reduced which also reduces the number of computations required to perform the larger convolutional layers. The Inception module can be viewed as a network inside of a network whose goal is to optimize layer topology.

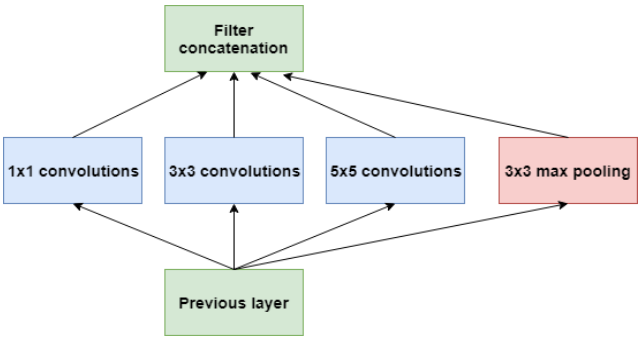


Fig. 3 Inception module structure.

D. Explainability

Despite the success of CNN and their improved architectures such as Inception and ResNet [6], the internal mechanisms of what causes the high classification accuracies remain unclear [7]. Substantial interest has been given to improving the classification accuracies of neural networks without focusing on the explainability of the inner layers. CNNs are commonly being used as black boxes where a structure is defined and then trained over large data sets. The emphasis placed on the resulting model is typically focused on achieving high test accuracy while forgoing transparency of the algorithm. Neural network interpretability is a burgeoning area of research with a few different approaches through feature visualization to better understand the inner workings of the neural networks. The imagery generated during feature visualization provides a means to provide human interpretability for neural networks. A better understanding of the internal networks may lead to: 1) Improved trust of CNN to perform safety critical tasks 2) Opportunity of retraining the network during a mission as new information is gained such as objects of interest 3) Improvement in CNN design by understanding which features and layers are most important for classification and which are superfluous.

IV. Feature Visualization Approaches

This section shows the output for different feature visualization techniques for the input image shown in **Fig. 4**.

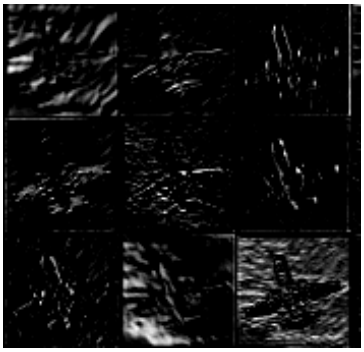
Fig. 4 Input image to visualization algorithm [2]**.**

A. Activation

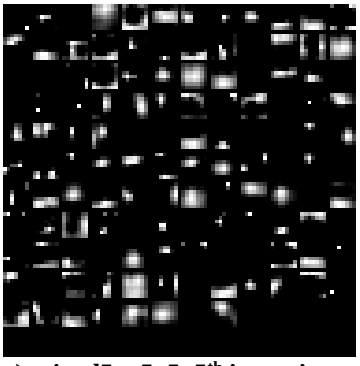
Activation is a primitive type of feature visualization in which the outputs for a given layer are visualized. This is most useful for the first few layers as they more directly map to image space, meaning objects still maintain their morphological appearance. The first layers find edges and contours. The deeper layers are more difficult to conceptualize because each neuron represents a combination of neurons from previous layers. Therefore the later layers appear more abstract because they no longer directly associate with 1 pixel – rather they represent a combination of filters on an image. **Fig. 5** shows a progression of activation visualizations from three select layers of the Inception network when the image from **Fig. 4** is input into the network.

Fig. 5a shows the activation of the 1st convolution layer. The intensity of each pixel directly corresponds to one convolutional filter. At this level, it would be possible to identify the exact filters that are applied on the image. For instance, one filter could be activating on vertical lines in a 7x7 kernel. **Fig. 5b** shows the activation of the 3rd convolution layer. The activations here are a combination of the two convolution layers before it. While the outline of the plane can still be seen indicating that the combination of filters to this point is activating on this region of the image, it is harder to identify the shape or texture that the network is activating upon. **Fig. 5c** shows the 5x5 convolution layer of the 5th Inception module which is towards the end of the network. At this layer, there is little to no explainable information that can be gleaned from visualizing the activations.

a) 1st convolution layer



b) 3rd convolution layer



c) mixed5a_5x5, 5th inception module

Fig. 5 Visualization of neuron activations for selected layers.

B. Deconvolutional Approach

Deconvolution provides a means of visualizing layers of CNNs in image space. Deconvolution maps information from layer(s) back to reconstruct the input image. The first step is deconvolution is to feed an input image through a CNN and map all features. The second step passes a feature map from a given layer through all subsequent layers by unpooling, rectifying, and finally filtering. The activations that are not associated with the feature map for a given layer are set to zero prior to passing the feature map through the network. In this way, the features for a given layer from a given input image may be interpreted through all subsequent layers.

Visualization using deconvolution for three layers is shown in **Fig. 6** through **Fig. 8**. In **Fig. 6**, the visualization for the first convolution layer is shown. Each grid image represents a neuron. Each image in **Fig. 6** is a feature activation extracted from the input image. At early layers of CNNs, primitive features such as edges and textures are extracted. The images in **Fig. 6** show a profile of the airplane with lines at different angles or textures. Each image shows how much a neuron is activating for its particular convolution filter. This gives us insight on what the particular neuron is activating on. For example, the bottom left image may be looking for sharp lines which only show up on the airplane but not the background. The top left image may be activating on low frequency changes which appear in the clouds as well as the airplane.

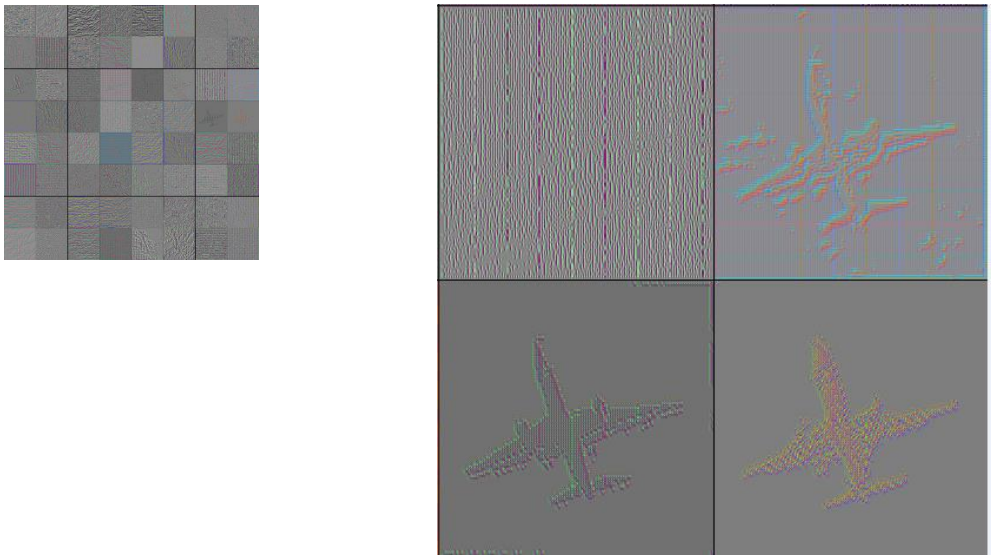


Fig. 6 Visualization of deconvolution reconstruction of 1st convolutional layer. The left image shows the entire layer while the right image shows 4 neurons from the layers.

The deconvolution output for the same layer as **Fig. 5**c is shown in **Fig. 7** below. Differing parts of the airframe, engines, and landing gear are emphasized by each neuron. Here, a neuron is a manifestation of neurons activated from previous layers. The appearance of the airframe at these layers shows a combination of lower level features. The focus of the neuron is shown to be the right wing in the left image, the cockpit in the top right image, and left wing in the bottom left. The neurons focus on different regions and substructures of the aircraft at this layer. Overall, the feature reconstruction shows a more clear structure for the airplane than in **Fig. 6** and **Fig. 5**c. Higher level features than the image primitives in the first layer are represented in the features activations for this layer in the deconvolution approach. The black boxes in **Fig. 7** represent switches and neurons that do not fire for the given input image. Theses switches do not pass information to the later CNN levels.

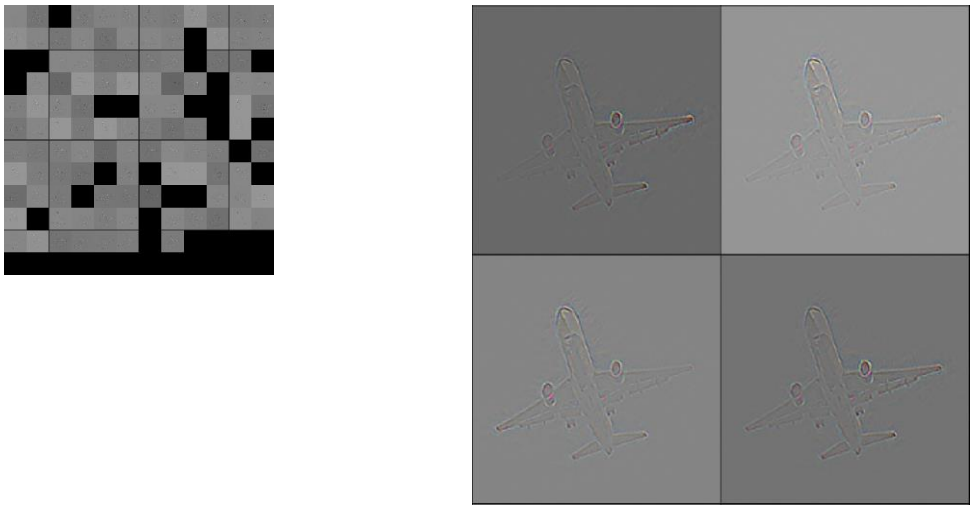


Fig. 7 Visualization of deconvolution reconstruction of (mixed5a_5x5, 5th inception module). The left image shows the entire layer while the right image shows 4 neurons from the layers.

Finally the feature activation in **Fig. 8** shows less variance between neuron output as the shape and contours appear more homogeneous between the images. The neurons in these layers emphasize features at the class level. The entire airframe, engines, landing gear, flaps and ailerons are represented by the features to provide the information needed to classify the object as an airplane. This is consistent with the hierarchical nature of the CNN, where later layers contain more complex features.

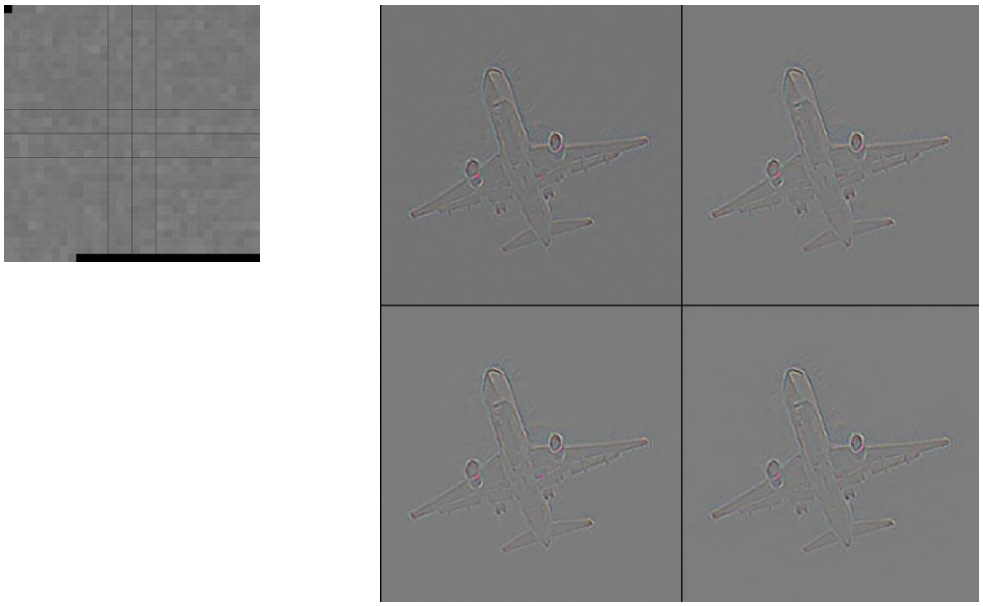
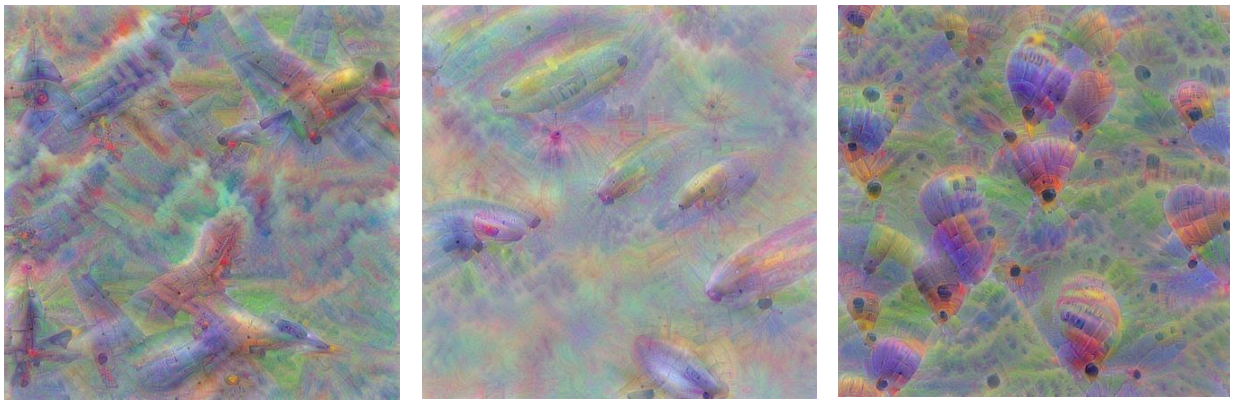


Fig. 8 Visualization of deconvolution reconstruction of (softmax 2 pre-activation). The left image shows the entire layer while the right image shows 4 neurons from the layers.

C. DeepDream

DeepDream [5] [8] is a method that modifies an input image such that patterns that stimulate a particular layer of a CNN are enhanced. The result are dream-like aberrations in the original image. Depending on the layer, this may amplify higher or lower level features. The DeepDream steps are:

1) start with an input image and choose a layer
2) extract the activations for the layer
3) set gradient to its activation
4) calculate gradient on image
5) revise image
6) repeat steps 2 through 4 for the number of iterations.



a) Airliner **b) Warplane/military plane** **c) Airship/dirigible**

Fig. 9 Visualization of DeepDream softmax2 pre-activation for 20 iterations from the Inception network. The input image for all three classes here is in Fig. 4. The output classes are specified in the captions for a), b), and c).

When performed at lower layers, texture information is inserted into the image. Higher layers such as those shown in **Fig. 9**, show higher layers of abstraction. At this layer, recognizable components of a class can be inserted into the image. While the input image was of an airliner, **Fig. 9b** and **Fig. 9c** show insertions of different classes. In this way, we force the network to use activations that do not match the input image. Places where the identifiable components appear provides a sense of the spatial interest in the image of the layer for a class that is not selected by the classifier.

DeepDream feature visualization may allow for correction of a network due to a problem from the training process. Suppose a network has only been trained on airplanes on runways. Then presented with an image of an airplane in the air, it may incorrectly classify because it was using features from the runway and airplane to classify planes. If we were to visualize the airplane class and see runways inserted into the image, then feature visualization may reveal this problem.

D. Mask R-CNN segmentation approach

The Mask R-CNN network extends the Faster R-CNN network by adding a segmentation pipeline. The additional pipeline performs pixel-based segmentation in parallel to the classification portion of the faster R-CNN. While segmentation by itself is not a feature visualization method, it is an improved method of inspecting the output of R-CNN, which aids in neural network interpretability. **Fig. 10** below shows output of a Mask R-CNN segmentation from a recent ATTRACTOR demonstration showing a Search and Rescue under the Canopy (SARUC) mission. In the demonstration, we tested multiple UAVs autonomously traversing a wooded environment while searching for people using a forward-facing camera. One of the capabilities needed for this mission is a person-detection algorithm. The Mask R-CNN algorithm's contoured segmentation is an improvement over the rectangular boxes of Faster R-CNN where the lines do not follow the edges of the classified object. Two people are correctly labeled and segmented in the image despite the occlusion of the person in the background by the other first person. The contour information of the classified people could lead to an operator trusting the algorithm because the shape is consistent with the classification result.

Fig. 10 Mask-RCNN showing detections of people. Mask-RCNN adds an object segmentation module on top of the region and classification modules of RCNN.

Another example mission is to confirm a landing zone is safe to land without people below [9]. **Fig. 11a** shows the output of Mask R-CNN from UAV imagery where people have been correctly identified. **Fig. 11b** shows a classification of a shadow as a human that resembles the profile of a head. Additionally, the trailer on the left is classified as a truck. The segmentation provides insight into the decision making process. The object confusion may be better understood by performing feature visualization. Visualizing the features for truck class may help understand why the trailer was labeled as a truck. The physical structure of trailers and trucks are similar with their boxy geometries, which may have been a factor with the misclassification. Reviewing the truck classification to increase the weight of features on the cab structure may help resolve this misclassification.

a) Three correct detections are shown with people in field at an altitude of approximately 100 ft. One person is missed by trailer on left. The false positive in the opening of the trailer resembles a person in profile.

b) Two false positives: a trailer is classified as a truck and shadow is classified as a person.

Fig. 11 Mask-RCNN on images showing detection of persons from a UAV-mounted camera.

V. Future Work

From the results we have generated, it is evident that the CNN weights and activations are easier to comprehend when projected into image space. This enables explainability of particulars parts of the network, which is needed for establishing a basis of trust. The algorithms we reviewed use a deconvolution or gradient ascent process to move backwards through the neural network and project activations to image space. Our plans in the future include exploring autoencoders for explainability and inference research. An autoencoder is a another type of neural network where the input data is decoded and re-encoded to reconstruct the same information that the input data represented, which results in a dimensionality reduction. An autoencoder's structure inherently has a forward and backward network with the goal of being able to reconstruct the original input. The encoded neurons can be viewed as a dimensionality reduction of the data set. Because the learned latent variables contain the information necessary to reconstruct the original images, visualization of the latent variables are expected to produce more distinct and discernable features compared to traditional neural networks. Variations of autoencoders are used to create generative models [10] and also for classification tasks [11].

VI. Conclusion

In this work, we reviewed several feature visualization techniques using aviation imagery. Each of the feature visualization techniques explores a different perspective on the internal layers: activation visualizes the weights, deconvolution visualizes the pixels from an input image that cause neurons to fire for a given layer, and DeepDream amplifies features and patterns within an image. Visualization of activation layers reveals image primitives at lower levels while higher layers are abstract. Deconvolution provides insight for the higher level features at deeper layers where the activation method is unsuccessful. DeepDream is useful for understanding the stimulation of a layer.

These image classification insights and rationale for that classification aid in explainability of autonomous decision making. Explainability is a critical aspect of establishing a basis for certification of autonomous systems. Building this basis by establishing metrics for trustworthiness and trust is the focus of the ATTRACTOR project. The context of a classification decision is useful to SARUC because it brings explainability to the autonomous system to make a decision such as: image this part of the woods more thoroughly because a shadow resembles a person. Or this part of the woods does not have humans, time to plan with the other agents performing the search for the next search location and pattern.

Understanding what caused the neural network to arrive at a decision provides a higher dimension of understanding the classification decision and adds explainability. Developing the means of contextualizing the classification decision will aid in the advancement of neural networks. As neural networks continue to outperform other methods for classification tasks, understanding why networks makes decisions is crucial to advancing the trustworthiness of networks to perform safety critical tasks and expand interactive capability alongside humans.

References

[1] F. Hohman, M. Kahng, R. Pienta and D. Chau, "Visual Analytics in Deep Learning:," *IEEE TRANSACTIONS ON VISUALIZATION AND COMPUTER GRAPHICS*, vol. 25, no. 1, pp. 1-20, 2019.

[2] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. Berg and L. Fei-Fei, "ImageNet Large Scale Visual Recognition," *IJCV*, 2015.

[3] S. Ren, K. He, R. Girshick and J. Sun, "Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks," in *Advances in Neural Information Processing Systems (NIPS)*, 2015.

[4] L. Tran, "Computer Vision for Precision Landing and Object Detection for Autonomous Package Pickup," in *AIAA Aviation*, Denver, 2017.

[5] A. Mordvintsev, C. Olah and M. Tyka, "Inceptionism: Going Deeper into Neural Networks," *Google Research Blog*, 2015.

[6] K. He, X. Zhang, S. Ren and J. Sun, "Deep Residual Learning for Image Recognition," in *CVPR*, Boston, 2015.

[7] M. D. Zeiler and R. Fergus, "Visualizing and Understanding Convolutional Networks," *CoRR*, vol. abs/1311.2901, 2013.

[8] C. Olah, A. Mordvintsev and L. Schubert, "Feature Visualization," *Distill*, 2017.

[9] P. Lusk and R. Beard, "Visual Multiple Target Tracking From a Descending Aerial Platform," American Control Conference, Milwaukee, 2018.

[10] D. Kingma and M. Welling, "Auto-Encoding Variational Bytes," in *ICLR*, 2014.

[11] Y. Li, Q. Pan, S. Wang, Peng Haiyun, T. Yang and E. Cambria, "Disentangled Variational Auto-Encoder for Semi-supervised Learning," *CoRR*, vol. abs/1709.05047, 2017.

[12] B. Vikani and F. Shah, *CNN Visualization*, 2017.

[13] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. C. Berg and L. Fei-Fei, "ImageNet Large Scale Visual Recognition Challenge," *International Journal of Computer Vision (IJCV)*, vol. 115, no. 3, pp. 211-252, 2015.

[14] Y. LeCun, Y. Bengio and G. Hinton, "Deep Learning," *Nature*, 2015.

[15] C. Szegedy, W. Lie, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke and A. Rabinovich, "Going Deeper with Convolutions," in *Computer Vision and Pattern Recognition*, Boston, MA, 2015.