Convolutional Neural Networks Can Achieve Human-Level Performance when Locating Critical Points in Textured Flow Field Images

James Enouen, Wen Fang, Jian Chen [1]

Computer Science and Engineering, The Ohio State University

Abstract

The field of visualization evaluation was founded on a key assumption that task abstraction is necessary to ensure the *validity* of the study. However, task decomposition can lose important contextual information making the transfer and integration of diverse experimental results for real-world uses challenging. Intriguingly, recent advances of neural networks seem to show that standard models can perform well in many real-world tasks [1, 2]. This immense behavior differences provide a plethora of opportunities in that integrating and combining tasks for CNNs may provide a more "realistic" measure to understand CNN capabilities and experience acting on the real-world tasks, that are complementary of those in human experiments. In this work, we present results from a convolutional neural network (CNN)-based study to replicate and compare its performance to two tasks humans performed in the user study of Laidlaw et al.: (1) locating all critical points in an image (localization) and (2) identifying critical point types (recognition); and shows that our simple CNN is able to achieve near perfect accuracy. These results help inform our long-term goal of integrating human and CNN expertise and understanding the limitations and preferences of modern day convolutional neural networks.

Keywords: CNN, cross-task evaluation, visualization tasks, vector field

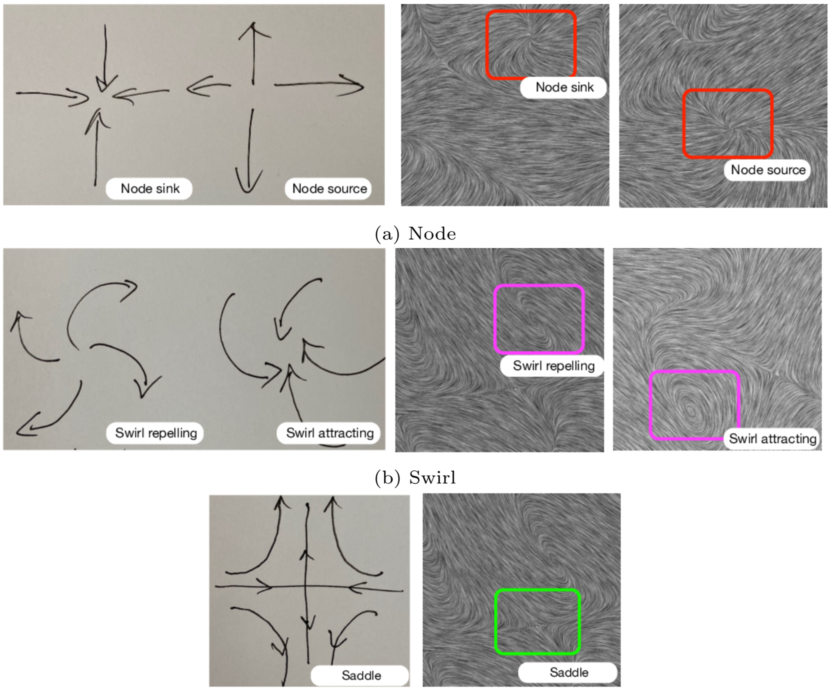
[1] Emails: {enouen.8, fang.661, chen.8028}@osu.edu

1. Introduction

A primary goal of scientific visualization is to precisely and accurately reveal the underlying physical phenomena through visual encodings. Decades of fundamental research attempting to understand what, when, why, and how to design visualizations have enabled systematic studies and creative design solutions which tout themselves as easy to use or designed with users' needs in mind. For the most part, validation of user experiences has used the same basic principles and designs for the past decade: hierarchical task analyses or task decomposition methods [3] to arrive at a set of universal tasks suitable to directly compare techniques. Often, it is found that different presentations of the same data and information best support different tasks and there perhaps does not exist a silver bullet to all tasks [4, 5]. Consequently, it is impossible to design effective visualizations without first considering the tasks for which the visualization will be used [6]. Despite that this decomposition or abstraction process is itself exceptionally challenging, task-specific advantages and evaluation methods from visualization solutions pertain to nearly all examples coming from complex, real-world scientific visualization evaluations [].

Taking the first serious consideration of the vector field valuation of scientific data as an example, Laidlaw et al. carefully compared six visualization approaches for the three vector flow tasks of: location of the critical points, critical point types, and the vector field advection [7]. Ware suggested a more complete set of six tasks, such as the advection trajectory, judging the speed, orientation, and direction at an arbitrary point, and the extreme speed and vorticity and turbulence [8]. In real-world uses, users are unlikely to switch between visualizations to choose the best solution. As a result, techniques suitable for *cross-task* conditions would be ideal. For example, viewers may *locate* the critical points and *see* the types all together.

Recent remarkable performance enabled by convolutional neural network (CNN) models have revolutionized many critical computer vision tasks. Many of these models are able to train on a large variety of input sets with minor



(a) Node

(b) Swirl

(c) Saddle

Figure 1: We group source and sink as node and we group both swirl repelling and swirl attracting together as swirl because LIC texture does not support orientation. For classification, we also introduce the class of no critical point present. Red denotes a node point, magenta denotes a swirl point, green denotes a saddle point, and blue color will denote no critical point.

modifications. For example, VGG [9] and Inception [10] family networks are commonly used in diverse sets of vision tasks and the testing results might be more sensitive to the quality of the training data rather than the tasks. This observation may lead to an interesting question of how well CNNs can perform on certain visualization tasks, especially the cross-task conditions when two elementary tasks are combined. Our vision is that in the future CNN and humans will collaborate, rather than merely humans' controlling or supervising CNNs. To realize this vision of human-CNN collaboration, CNNs must achieve good theoretical performance in order to be a good partner for humanity. As a result, we must test CNN's expertise on tasks in terms of their accuracy and biases.

In this work, we examine vector field visualizations using a CNN model to test its accuracy on the texture-based visualization of line integral convolution (LIC), one of the six vector field visualization methods measured in the seminal work of Laidlaw et al. [7]. We choose LIC because we first believe that ImageNet-based CNNs prefer textures patterns [11] and thus it is likely texture-based methods are most effective. Second, humans attain a considerable accuracy using texture-based visualization for the localization task of locating all critical points in a vector field image [7]. Consequent of this choice, it would be unfair to ask orientation questions of the direction of the node, saddle or swirl because the LIC texture does not carry this information. Performing the same tasks as Laidlaw et al. using CNNs allows us to test an optimal case scenario prior to a large-scale study to examine other canonical visualization methods through integrating two visualization tasks of localization and critical point type together. Finally, testing the same visual stimuli using CNNs allows us to directly compare the expertise of human and CNNs.

Our current results support that our CNN can achieve near perfect accuracy for identifying critical point types when the model is trained on critical point type tasks and 81% accuracy using the same model for localization task.

2. Related Work

This section reviews work which has influenced ours in the areas of vector field visualization and CNN methods for visualization tasks.

2.1. Vector Field Visualization and Evaluation

Vector field visualizations have proven to be very useful to explore, analyze, and gain insights into complex physical phenomenons [12]. The success of many approaches depends on humans' visual intelligence to decipher visual stimuli from the increasingly complex and heterogeneous data often coming from simulations and modelings [13, 14, 15]. Often, empirical studies produce predictions about real-world task performance by decomposing and abstracting real-world problems into testable tasks. Experiments involving systematic variation of parameters have been achieved by carefully choosing and controlling independent and dependent variables of the visualizations for testable tasks. Another approach is to perform head-to-head experiments to compare concrete visualization methods. For example, Laidlaw et al. compare six classical two-dimensional (2D) vector field visualization methods in a head-to-head experiment: GRID (icons on a regular grid), JIT (icons on a jittered grid), LIC (line-integral convolution), OSTR (image-guided streamlines), and GSTR (streamlines seeded on a regular grid) [7]. These methods are chosen by their various pattern-revealing abilities, such as contours, shapes, and sampling methods etc. Furthermore, the study compared the time and accuracy of the viewers' responses over all of the different visualizations. Some of the conclusions they came to regarding LIC are as follows: when locating critical points, the users took less time and were more accurate in vector field images generated by the methods not explicitly laid on a grid and techniques such as LIC, OSTR, and GSTR; when identifying the critical point types, the LIC method did not help with accuracy, possibly because the method does not show the orientation of the vector field. In this work, we combine the categories and ask the network both localization and identification tasks simultaneously while keeping mind of the visual stimuli provided to the network.

2.2. CNN for Visual Recognition and Detection Tasks

The other main component of this work is the use of machine's intelligence such as CNNs to perform human-level visualization tasks. Visual question answering (VQA) tasks involve getting a neural network to answer a question given a visual stimulus. In this same line of work, Haehn et al. evaluated the 'graphical perception' of a variety of CNNs by evaluating their performance on different tasks on bar charts and pie charts [16]. CNNs performed better than humans when asked to estimate quantities directly from visual marks of bars. CNNs were not able to compute ratios between two data items depicted on bars; however, the researchers noted that with proper training it is possible that CNNs would perform better on these tasks. By comparing different CNN models, Haehn et al. also found that the VGG was consistently better than others in different tasks due to its ability to better anti-alias the input and feature map signals. Additionally, the authors claimed that training the weights of the network from scratch would be a better strategy to build a CNN that works for specific visual query tasks [16].

Here we follow Haehn et al. by measuring accuracy for visualization tasks by replicating the seminal work of the Laidlaw et al. in the scientific visualization domain. Our work will branch away from the information visualization which Haehn et al. study in their work which has many arbitrary choices like where to place the bars in a chart and instead focus on the scientific visualization of a vector field. Additionally, because CNNs depend on repeated applications of convolutional filters, we hope to leverage the fact that a texture image like LIC may better utilize texture stimuli rather than the shapes in Haehn et al. We suspect that this texture stimuli will allow CNNs to achieve better accuracy in vector field visualizations.

Geirhos et al. are among the first to understand the CNNs' ability to interpret fundamental visual stimuli of texture and shapes [11]. Their study represents a clever setup of the combined texture-shape stimuli to study CNNs' preferences and their work demonstrates that network architectures trained with texture-based representation on ImageNet would prefer textures and "stylized-ImageNet", a stylized version of ImageNet allows an interesting shape-based representation. These results convincingly suggest that unexpected emergent biases of networks as well as improving the network robustness by integrating humans' shape recognition abilities. Similarly Wurster et al. suggests that incorporating human gist processing (human visual intelligence) and CNN intelligence (machine intelligence) improves cancerous tissue screening in mammograms [17]. To achieve our long-term goal of making humans and CNN efficient partner, we attempt to begin to study the CNN's ability to read visualizations to evaluate whether or not CNNs can achieve impressive accuracy by asking the network to do a high-level task which combine the localization and the critical point type tasks in Laidlaw et al. [7].

The advantages of texture stimuli have suggested its important role for object recognition in standard CNNs, unlike humans who rely on shape stimuli. Standard CNNs are bad at recognizing object sketches from shapes when texture stimuli are missing. Additionally, texture-enabled local information are "salient" enough for CNNs to "solve" ImageNet object recognition when a linear classifier on top of a CNN's texture representation (Gram matrix) to achieve near-perfect linear classifier. Furthermore, local texture patches rather than global object parts from shapes revealed surprisingly effective methods for shape classification. Taken together and in the light of these findings, we believe that texture stimuli may be important for detection these fundamental research mainly in vision science literature, we suspect that CNNs would be able to utilize texture stimuli that have been successful in visualization to answer some recognition and detection questions.

3. Methods

Our goal in this work is to perform representative vector field visualization perceptual tasks using CNNs to locate critical points and identify critical point types from a LIC visual encoding. In this section we outline the core elements of our study design and procedure. Data, code, and supplemental materials are available in our project repository.

3.1. Experimental Configurations

Datasets.. In order to assess texture usefulness to CNN, we conducted an experiment with the only differences being the tasks between the CNN and the human experiment. We have used the same rendering methods for CNNs as the human experiment in Laidlaw et al. [7]. To accomplish the training and testing of the CNNs, we required a controlled set of stimuli. The algorithm used to generate the vector fields was performed using 2D radial basis function interpolation of a 2D vector field. This representation was then used to generate a gridded, discrete representation of the vector field. This was then converted into each of the vector field visualization methods we utilized. In this process we generated approximately 25000 two-dimensional vector field data sets with a resolution of 400 by 400. Additionally, the critical point information was computed for each image. Among this data, we picked a balanced set with 6000 saddle, 6000 node, and 6000 spirals We then further split this data into a balanced training, validation, and testing set. We used a 60-20-20 split with 3600, 1200, and 1200 points of each type in each corresponding set.

We then created the corresponding LIC visualizations by first normalizing the vector field and generating white noise as the input texture. Next, a line integral convolution was applied by advecting a particle through the field to generate flow imaging. Finally, an intensity mapping was applied to correct the loss of contrast due to the convolution [18].

Because the LIC method is unable to depict the flow direction without an extra visual channel, it is theoretically impossible for a CNN to differentiating a source from a sink, or a repelling-spiral from an attracting-spiral. Consequently, we considered each of these pairs to be one class as previously mentioned.

3.2. Neural Network Architecture

Classification. We originally treated the problem in the object-detection framework, but the size of the network made it very difficult to propagate error throughout this large network and train a full end-to-end system. We then adapted the problem to classification by only feeding the network small (47x47) patches of the entire (400x400) visualization and asking the network to classify the patch as not a critical point or give the type of critical point. The model used was a slight adaptation of the VGG architecture to create our four-way classifier which entailed six convolutional layers followed by three dense layers. This model was significantly easier to train as the loss easily propagated through the entire network. The network was trained for 50 epochs and the model with the least validation loss was taken as the final model. The training hyperparameters were: a learning rate of 1e-3, exponential decay of $(0.9)^{floor(epochs/10)}$, optimizer of SGD, loss of categorical cross-entropy. Again, the model will be available in our repository.

Localization. After we trained this classification network, we were able to apply the model as a patch-based classifier. This was then combined with very simple computer vision techniques (thresholding and morphology) to get prediction regions for critical points. First, a softmax volume, depicted in Figure 2, was generated from the application of the classifier to each patch of the LIC image. Second, each of the critical point softmaxes were thresholded to yield prediction regions. Next, regions with less than 30 pixels were removed as predictions and the final regions correspond to the network's final decision as shown in Figure 3.

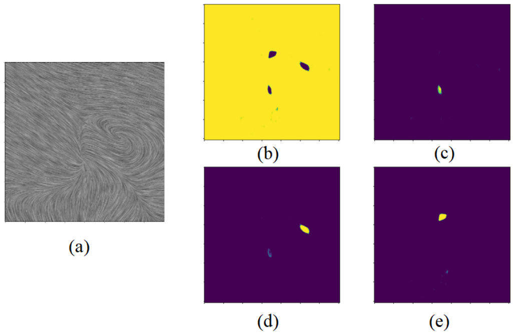


Figure 2: The 354 x 354 x 4 volume generated by the softmax scores of each patch (where 354 = 400 - 47 + 1.) (a) is the original LIC texture image. In the other images, yellow denotes values close to 1.0 and dark blue denotes values close to 0.0. (b) is the none softmax value; (c) is the node softmax value; (d) is the swirl softmax value; (e) is the saddle softmax value. We can see the network believes there is no critical point throughout most of the image except for the three holes corresponding to critical points.

4. Results & Discussion

4.1. LIC

Classification. We first go through some of the results which maintain the perspective that the network is a classification based network. We trained the model as described and achieved an impressive testing accuracy of 98.16%. If we look at the ROC curves in Figure 4, we can see that the model does an extremely good job of distinguishing critical point types with AUC values of: none 0.9994; node 0.9975; swirl 0.9983; saddle 0.9997. Again in Figure 4 we

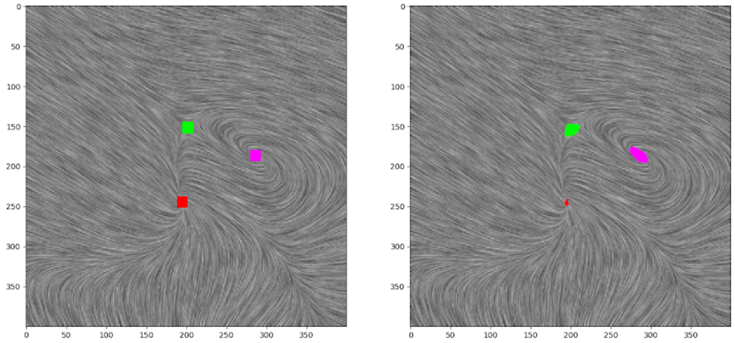


Figure 3: In this figure, we can see that the above softmax scores led to the correct predictions of each of the three critical points. The ground truth locations (left) show the critical points in the same locations with the same types as the network predictions (right)

can see in the zoomed section that there is slightly worse performance for the classes node and swirl.

We can reinforce this belief that the node and swirl critical points are the most difficult by looking at the confusion matrix in Figure 5 where we see that the two most common mistakes are calling a node a swirl and vice-versa. To illustrate how difficult these examples are, we provide some of these misclassified patches in Figure 6. It is surprisingly difficult to guess which class each example belongs to because of how visually similar these two classes can be. Overall, these very high AUROC and accuracy scores indicate success of the CNN and can be attributed to the texture-based format of LIC imaging being well suited for CNNs; however, it should still be kept in mind that these impressive results are still under the classification regime.

Localization. As described in the methodology, the classification network is then used to determine a prediction region on the full image. Figure 7 provides examples of the final prediction results to see what kind of mistakes the network makes. All of these results will have the ground truth overlaid on the LIC visualization on the left side and the network prediction overlaid on the visualization on the right side. Each of the three colors corresponds to the three types of critical points as throughout the work. We can see that visually the model does

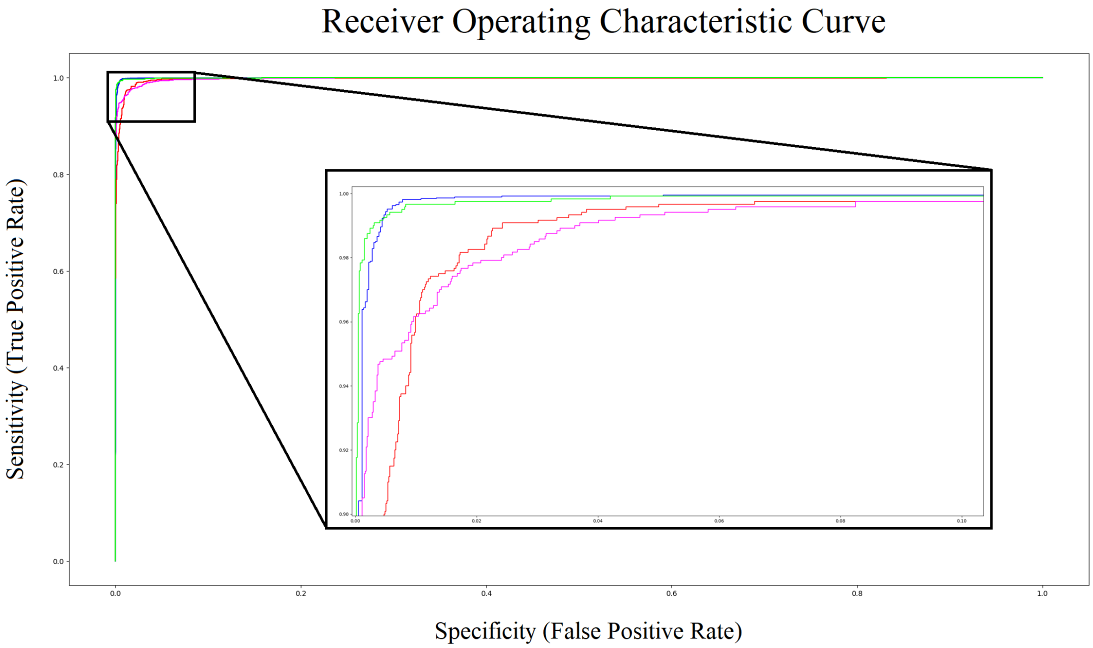


Figure 4: ROC curve for all classes. blue: none, red: node, magenta: swirl, green: saddle.

a fairly good job of locating and identifying critical points.

Numerically, we actually have a 100% recall in the test set, indicating every critical point of the vector fields was spotted by the network. However, we only have a precision of 81%, meaning that we had a number of false positives lowering our accuracy. The primary cause is most likely the combined effects of:

- Using a classification network to perform a localization task
- Using a training dataset which had equally distributed examples of non-critical points

These two items together lead to the network being susceptible to areas which looked vaguely like critical points and insufficient training stimuli led the network to assume these regions to be critical points. If the network were trained with more of these borderline examples or used localization loss to disincentivize

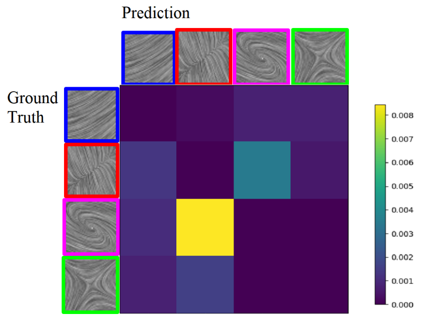


Figure 5: Multiclass Confusion Matrix for Classification Network. Again, we have that blue is no critical point, red is a node point, magenta is a swirl point, and green is saddle point. Each row corresponds to the ground truth of the testing patch and each column corresponds to the predicted label by the classifier. We again use the 'viridis' colormap where yellow corresponds to the highest value and dark blue corresponds to zero. Because the classifier is nearly perfect, we additionally zeroed out the diagonal to make the confusion pattern visible.

these predictions, it would be able to discriminate these difficult examples as non-critical points. While this unfortunately causes the network to only have human performance on the localization task, it is clear how to improve the CNN's performance on this task.

4.2. Comparison of Performance Results to Humans

As mentioned, we can attribute a lot of the network's success to how well set up the texture-based LIC method is for convolutional neural networks. As can be seen in Figure 8, the CNN vastly outperformed humans on all visualization methods in identifying the critical point type. Additionaly, only achieving human-level performance on the localization task as seen in Figure 8 we believe to be a resolvable issue by using a CNN better equipped for the localization task. Overall, it is clear that CNNs can achieve human-level and better performance on these visual tasks.

4.3. Limitations & Future Work

Akin to Haehn et al.'s study to replicate the Cleveland and McGill's study, we shall ultimately run a set of experiment to repeat those of Laidlaw et al.'s set of experiments by adding more visualization methods. We will also use the modern CNN architectures, e.g., VGG-19 and Inception-4 to choose the most

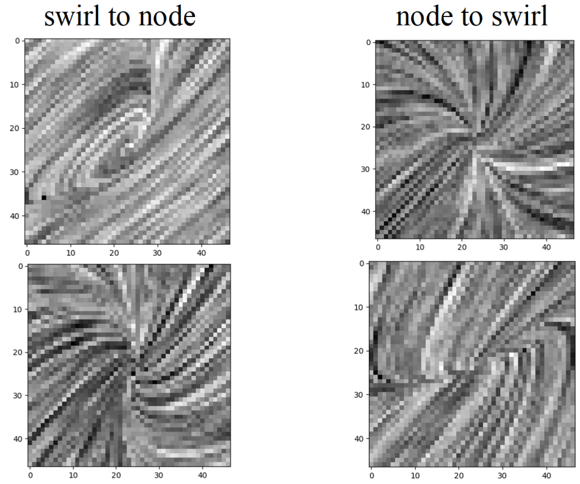


Figure 6: The left column is examples which are actually swirls and the right column is examples which are actually nodes

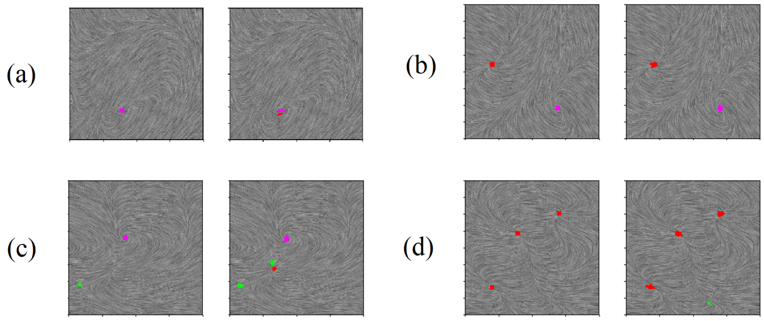


Figure 7: Final Prediction Results

suitable architectures for testing the visualization methods, which have achieved outstanding performance on computer-vision tasks for the past two decades. This more extensive set of experiments will allow us to further investigate the limitations and preferences of CNNs as well as differentiate their preferences from those of humans. If the ultimate goal is to replace the user study with the neural network study, it is necessary to understand the interaction of preferences between humans and CNNs in visual queries. Regardless, even with this simple CNN, we show that we can achieve a considerable accuracy gain compared to humans. As a result, our work contributes to understanding the ability of CNNs

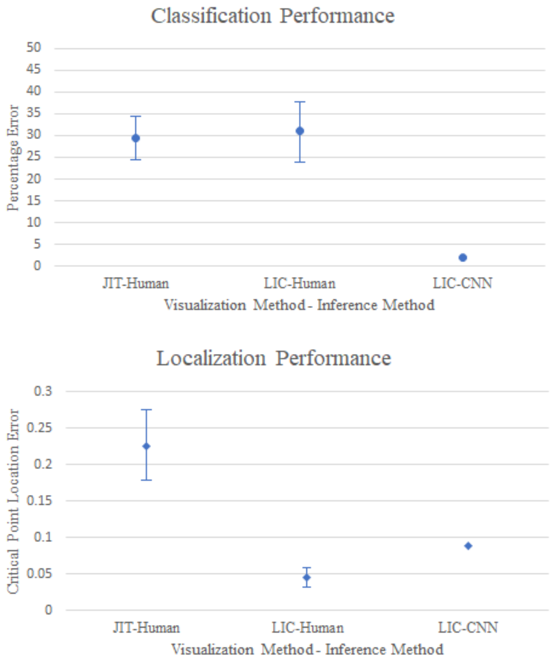


Figure 8: Comparison to human performance

to perform relatively complex tasks.

5. Conclusion

This work is the first step in using CNNs to perform visual tasks on scientific visualizations. Our results achieved and surpassed human-level performance on critical point classification using the LIC visualization. Because CNNs are able to complete these visual queries, they could have interesting applications in future user studies and graphical perception tasks. It is yet unclear whether CNNs will have the same preferences as humans in their visual representations, but we have already seen CNNs can outperform humans on specific tasks.

References

[1] K. He, X. Zhang, S. Ren, J. Sun, Delving deep into rectifiers: Surpassing human-level performance on imagenet classification, in: Proceedings of the

IEEE international conference on computer vision, 2015, pp. 1026–1034.

[2] L. Gatys, A. S. Ecker, M. Bethge, Texture synthesis using convolutional neural networks, in: Advances in neural information processing systems, 2015, pp. 262–270.

[3] T. Munzner, A nested model for visualization design and validation, IEEE transactions on visualization and computer graphics 15 (6) (2009) 921–928.

[4] T. Isenberg, P. Isenberg, J. Chen, M. Sedlmair, T. Möller, A systematic review on the practice of evaluating visualization, IEEE Transactions on Visualization and Computer Graphics 19 (12) (2013) 2818–2827.

[5] H. Lam, E. Bertini, P. Isenberg, C. Plaisant, S. Carpendale, Empirical studies in information visualization: Seven scenarios, IEEE transactions on visualization and computer graphics 18 (9) (2011) 1520–1536.

[6] B. Shneiderman, The eyes have it: A task by data type taxonomy for information visualizations, in: Proceedings 1996 IEEE symposium on visual languages, IEEE, 1996, pp. 336–343.

[7] D. H. Laidlaw, R. M. Kirby, C. D. Jackson, J. S. Davidson, T. S. Miller, M. da Silva, W. H. Warren, M. J. Tarr, Comparing 2d vector field visualization methods: a user study, IEEE Transactions on Visualization and Computer Graphics 11 (1) (2005) 59–70. `doi:10.1109/TVCG.2005.4`.

[8] C. Ware, Information visualization: perception for design, Elsevier, 2012.

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

[10] C. Szegedy, S. Ioffe, V. Vanhoucke, A. A. Alemi, Inception-v4, inception-resnet and the impact of residual connections on learning, in: Thirty-First AAAI Conference on Artificial Intelligence, 2017.

[11] R. Geirhos, P. Rubisch, C. Michaelis, M. Bethge, F. A. Wichmann, W. Brendel, Imagenet-trained cnns are biased towards texture; increasing shape bias improves accuracy and robustness, International Conference on Learning Representations (ICLR).

[12] R. S. Laramee, H. Hauser, H. Doleisch, B. Vrolijk, F. H. Post, D. Weiskopf, The state of the art in flow visualization: Dense and texture-based techniques, in: Computer Graphics Forum, Vol. 23, Wiley Online Library, 2004, pp. 203–221.

[13] J. Sanyal, S. Zhang, G. Bhattacharya, P. Amburn, R. Moorhead, A user study to compare four uncertainty visualization methods for 1d and 2d datasets, IEEE transactions on visualization and computer graphics 15 (6) (2009) 1209–1218.

[14] W. Chen, S. Zhang, S. Correia, D. S. Ebert, Abstractive representation and exploration of hierarchically clustered diffusion tensor fiber tracts, in: Computer Graphics Forum, Vol. 27, Wiley Online Library, 2008, pp. 1071–1078.

[15] S. Marchesin, C.-K. Chen, C. Ho, K.-L. Ma, View-dependent streamlines for 3d vector fields, IEEE Transactions on Visualization and Computer Graphics 16 (6) (2010) 1578–1586.

[16] D. Haehn, J. Tompkin, H. Pfister, Evaluating graphical perception with cnns, IEEE Transactions on Visualization and Computer Graphics (IEEE VIS) to appear (X) (2018) XâX.

[17] S. W. Wurster, A. Sitek, J. Chen, K. Evans, G. Kim, J. M. Wolfe, Human gist processing augments deep learning breast cancer risk assessmentarXiv:1912.05470.

[18] B. Cabral, L. C. Leedom, Imaging vector fields using line integral convolution, Tech. rep., Lawrence Livermore National Lab., CA (United States) (1993).