Insights and applications from data driven representation learning

by

Baladitya Yellapragada

A dissertation submitted in partial satisfaction of the

requirements for the degree of

Doctor of Philosophy

in

Vision Science

in the

Graduate Division

of the

University of California, Berkeley

Committee in charge:

Professor Stella X. Yu, Co-chair
Professor Bruno A. Olshausen, Co-chair
Professor Martin S. Banks
Professr Alexei Efros

Fall 2021

Insights and applications from data driven representation learning

Abstract

Insights and applications from data driven representation learning

by

Baladitya Yellapragada

Doctor of Philosophy in Vision Science

University of California, Berkeley

Professor Stella X. Yu, Co-chair

Professor Bruno A. Olshausen, Co-chair

This dissertation is an exploration of data-driven discovery, inspired by neuroscientific studies of the brain. Each of the three projects listed will describe a different domain of input data (self-driving video, medical image, and biological audio), and how investigating neural network behavior trained on that data can reveal insights for each underlying task. Chapter 2 assess motion selectivity in a self-driving network trained to predict two output tasks: steering and motor. We show how different control conditions can define temporal behavior, as well as how frame order is only implicitly learned if relevant for the task, even if the frame order is present in the input and output training data. Chapter 3 assesses self-supervisedly learned representations from retinal fundus images. We show how these learned representations can drive a voting scheme classifier to match supervised and human expert baselines for disease severity prediction in this field, minimizing the bias enforced from clinically relevant ground truth labels. These representations can be further probed to discover mislabeled or easily confused data, as well as phenotype groupings in retinal images that pertain to other pathology and physiology of the subject. These imply NPID and cluster analysis tools could aid clinicians organize and label data from multiple tasks, an expensive process that requires uncommon expertise. Similarly, Chapter 4 extends this idea about data-driven learning to the audio domain. Here, self-supervisedly learned representations from zebra finch data yielded feature encodings that were functionally relevant for classifying vocalization calls, driving a voting scheme classifier to match supervised baseline performance on a generally difficult task of intra-species audio discrimination. We convert audio waveforms to spectrogram image representations of sound signals, and train a CNN on these inputs, so we can probe these visually-defined audio features. To do so, we assessed how neuronal behavioral preferences can be described by a mid-level representation space of audio (the modulation power spectrum), as well as how these features compare to mid-level audio features correlated with zebra finch brain activity. Data-driven algorithms can learn representations with

minimal bias, so commonalities between artificial and biological neural systems imply similar encodings are optimally learned. All in all, this dissertation has evaluated deep learning applied on a host of real world tasks aside from standard datasets curated for computer vision. Though each project requires a different lens for explaining functionally salient behavior, we offer data-driven insights into each underlying task that seem to be consistent with experimental findings in neuroscience and medicine.

To Narayana Yellapragada and Bhavani Mamidanna

For patiently supporting me through this explorative period in my life, while expressing genuine interest in what I was studying and researching.

Contents

List of Figures

Acknowledgments

None of these projects in the dissertation would have been possible without the support of many people through various stages of my graduate experience.

First and foremost, I would like to thank my parents, Narayana Yellapragada and Bhavani Mamidanna, whose emotional and financial support carried me through many ups and downs over the past 8.5 years. My sister, Aparna, who always knows how to stay positive, was also instrumental at shaping my livelihood as a graduate student. With respect to non-immediate family, the life discussions I have had with Srijna, Harita, Eshwar, Rishabh, Shashank, Ramya, and Sid, have helped me grow as an adult and deal with issues in grad school I had not experienced before.

Outside of family, I would like to thank specific members of the Vision Science community for either helping me get into the Vision Science PhD program, helping me find academic projects that suit my interes, or helping me navigate through campus administration: Jack Gallant, Martin Banks, Bruno Olshausen, Michael Silver, Jitendra Malik, Carissa Caloud, and Kaitlyn Guthrie. Additional thanks to Alexei Efros for being the outside member of my thesis committee.

I am also grateful for having had the opportunity to collaborate with several professors for the projects that led to this dissertation. Karl Zipser, Glenn Yiu, and Frederic Theunissen helped hone my research skills at applying deep learning to various domains and investigating novel representations that are expertly relevant to each domain. Stella Yu provided me with much needed focus alongside my constantly expanding research interests and helped me prioritize publishable beginnings, ends, and narrative flows for each academic pursuit; my graduation schedule was only possible because of her pushes.

I appreciate the collaborators I have worked on projects with, both those with whom I have and have not published papers. Specifically, I thank Bill Sprague, Sylvain Reissier, Yiqi Hou, Tushar Pankaj, Sauhaarda Chowdhuri, Alexander Anderson, Arian Ranjbar, Qian Yu, Zhirong Wu, Ziwei Liu, Rudrasis Chakraborty, Aaron Reite, Scott Kangas, Zhongqi Miao, and Nils-Steffen Worzyk. In addition, I would like to thank other members of labs I have been associated with for cultivating a health appreciation for academic discussion; these researchers are Steven Cholewiak, Agostino Gibaldi, Guy Isley, Dylan Paiton, Charles Frye, Yubei Chen, Brian Cheung, Xudong Wang, Zhihang Ren, Jiayun Wang, Chun-Hsiao Yeh, Jesper Christensen, Jyh-Jing Hwang, and especially Tsung-Wei Ke. I would like to make a notable call out to Sascha Hornauer, who has been an extraordinary collaborator, mentor, friend, and lifesaver at crucial moments like for this dissertation submission; truly most of my graduate experience was shaped by his guidance, and I gained much inspiration for applied deep learning through his works.

Within the Vision Science and Computer Vision community at UC Berkeley, I want to shout out to Vasha Guerin Dutell, Joel Bowen, Lauren Spano, Sahar Yousef, Mayur Mudigonda, Pulkit Agrawal, and Mark Lescroart for helping me learn that graduate school is more than just research and navigate through unexpected stressors. I also want to thank other members of my class, like Elizabeth Lawler, Sanam Mozaffari, Angelica Godinez, Billie

Beckwith Cohen, Jazzi Junge, Sarah Singh, and Ethan Bensinger for making me feel welcome in my cohort.

With regards to personal friends, I would like to thank Adrian Tzonev, Ann Dam, Sachin Rangarajan, Sravani Tammiraju, Praneet Wadge, Peter Nakamoto, Jared Porter, Ritu Kiragi, Arunava Sarma, Rohan Ramakrishnan, Chetan Vakkalagadda, Aaron Lin, Kunal Modi, Amaresh Shukla, Nikhil Raghuram, and countless others who I would love to mention if I had the space. You kept me sane this whole time.

Finally, I want to thank Baama (aka, Suryakumari), my grandmother, for patiently waiting for me to graduate before I visit her in India again; this pandemic has not made it easy. In addition, I want to thank my uncle Venkata Narayana Rao ”Bhaskar” Mamidanna for teaching me the value of both being there for my family and being there for myself; I will forever miss you, ”budday”.

Chapter 1

Introduction

1.1 Importance of CNN Representations

A benchmark of human vision is the ability to encode world views through generalizable representations. Structured patterns can be learned and related across objects and scenes, often guided by an unclear level of supervision. There is evidence that this representation learning is also relevant for deep learning on computer vision tasks. Convolutional Neural Networks (CNNs) can do more than magically solve tasks they are trained on. Specific filters do not need to be forcibly learned, but will naturally emerge from learning on different data with a similar domain of image statistics. Similar representations can even be learned across different tasks or levels of expert label guidance. The focus of this dissertation is to draw a parallel between neuroscience and CNN feature interpretation experiments, and to reveal how understanding learned CNN representations can give underlying insights into a variety of applied tasks.

1.2 CNNs for Visual Learning

Deep learning algorithms have been been at the forefront of visual learning tasks this past decade. Compared to other machine learning approaches (Support Vector Machines [76], linear classifiers aided by Gaussian Mixture Models [95], decision trees [3], ensemble models [10], and SIFT-based classifiers[144]), CNNs have dominated on the accuracy leader boards for object classification of ImageNet data since 2012[76, 106, 72]. Deep learning approaches were subsequently tailored for other research tasks, like object detection on MS COCO data (Fast R-CNN)[52] and semantic segmentation on PASCAL VOC data (R-CNN)[57].

CNNs for Object Recognition

Deep CNNs can be trained to perform well on visual tasks because they learn patterns across hierarchically complex scales of representations [149]. Earlier filters identify low-level concepts such as color, edges, and curves, and later layers focus on higher-level features such as animals or animal parts. Although CNNs are typically used for natural image tasks such as animal classification [106], aerial-view vehicle detection [140], and self-driving [13, 15], panoptic segmentation of scenes [69], etc. these algorithms have also been adapted for datasets and tasks that were not curated for computer vision learning (like biological applications). These will be discussed in a Chapter 4.

Feature Visualization for CNNs

Understanding these learned representations has been of common interest since around 2010 when neural networks gained popularity as the rise in computational power and memory made them more feasible for applied research [75, 11]. There are many methods available to probe neural filters across later layers of a trained network.

The first is gradient-based receptive field generation methods. These techniques aim to visualize the patterns that maximally activate a given or group of neurons. They iteratively update an input image by back-propagating a loss that maximizes activity for a target neuron. This can be achieved with random noise inputs through gradient ascent [11], or with maximal activating inputs through deconvolutional layers [141]. Regardless of technique, later layer visualizations are harder to interpret correctly. Improved variants aim to apply priors to clean up visualizations [88] or guiding a gradient through non-linearities without deconvolutions [114].

The second is image synthesis techniques. In the 80s and 90s, more naturalistic image synthesis helped neuroscientists and psychologists understand human perceptual groupings by modeling defined visual features. The most popular example is by Portilla and Simoncelli to define a model for texture synthesis [99]. Modern image synthesis relates to generative neural network, primarily generative adversarial networks (GANs) that can even provide a level of control for feature generation [53, 27, 58]. However, using one neural network to probe the representations of another network can be tricky to interpret generalizably as the first is also not easily explainable. A middle ground is image synthesis techniques that use a single network's gradient to guide random texture generation from a single base image in a way that maximizes activity for a given layer, like Ecker [50]. This method can also be considered a variant of gradient-based receptive field method for networks.

Another style of feature visualization is saliency map generation [114]. This highlights image areas from a given image that drive a given neuron's activation. This differs from generating a pattern from a random signal in that the input is already defined, but the feature needs to be interpreted from the visualization. Primarily, the easiest neurons to interpret are output neurons in supervised networks because their activations are directly tied to class-discriminable features. Class-activation maps or self-driving maps are the top use cases of this technique [148, 14].

The last style of feature visualization techniques is network dissection [8]. The idea is estimating neuron preferences by correlating output activity changes with quantified input features, across many input samples of varying feature quantities. This technique can be achieved with expensive, dense pixel-wise labeling for a given feature domain (e.g., texture, color, object, etc.). A simple application of this was for self-driving, to help determine which parts of an image drive a network to the left or drive. Bau, et al., have shown that this methodology can describe neuronal preferences for low level and high level feature representations, and that this hierarchy of features learned across artificial CNN layers parallels a hierarchy of features learned by biological cortical layers. Low level features like color and texture are preferred in early layers, and high level features like part, objects, and scenes are represented in layer layers.

1.3 Brain Representations

Inspiration from Neuroscience Experiments

Fundamentally, CNNs are predictive models that are motivated by brain structures, and the brain is another computational black box that learns representations. We can take inspiration from the frameworks that exist for studying biologically learned representations.

The main techniques from neuroscience or psychology are most closely tied to network dissection. The framework is to quantify input features and correlated differences in neuronal activity or neuronal correlate activity with differences in features across input samples.

These style of investigation is tied to predictive feature encoding. Such predictive models have helped researchers show evidence for a whitening pre-processing technique in the retina that allows for uniform activation of responses across an imbalanced distribution of input spatial frequencies [7]. Similar experiments have revealed differently oriented Gabor-like (i.e. sombrero-shaped, center-surround features) preferences for neurons in cortical V1 [118]; contours, colors, motion, and stereoscopic disparity for neurons in V2 [51, 17]; more complex contour, curvature, and color for neurons in V4 [87, 44, 91, 33, 43]; and faces or expertise for neurons in IT [138, 104].

Many of these biological experiments revealed insights about the underlying experience of the human visual system, and they were not localized to one layer of the brain. For example, the retinal whitening experiment validated previous statistical evaluations of images that there was an average 1/f distribution of spatial frequencies present in naturalistic stimuli [105]. Similarly, IT experiments showed that neuronal tunings sensitive to faces or expert-level recognition showed how ingrained socializing was with modern brain evolution of higher level processing.

Overall, there are many similarities between artificial neural network functionality and biological cortical functionality. Of note are two shared properties: (1) a parallel hierarchy of features from low to high in both neural systems, and (2) positional invariance of higher-level processing units in both systems. This implies an equivalent functionality between both brain feedforward hierarchy and neural network pooling layers. Of course, specific feature preferences at specific hierarchies may vary, or differences may exist between the brain and CNNs in response to particular visual situations (e.g., crowding). However, an overall shared hierarchy implies that many experiments that interpret the brain may also be viable for neural network research.

This dissertation aims to take inspiration from biological neural modeling experiments. Each project listed relates to feature exploration across a different domain of input data (self-driving video, medical image, and biological audio) to functionally describe learned features unique to each domain. The goal of this dissertation is to demonstrate that studying neural networks can reveal insights about what input representations are actually salient for accomplishing any underlying task.

Chapter 2

Motion Sensitivity Representations for Self-Driving

2.1 Introduction

Much of the feature interpretation research discussed in the Introduction has been on single image input tasks. One notable single image network was for a self-driving task [13]. Here, PilotNet, a CNN of a particular size (5 convolutional, 3 fully-connected layers) was trained to predict single steer commands for a fixed-speed imitation learning task on dashcam driving data. They generated saliency maps for steering prediction by averaging output activations across each convolutional layer (upsampling of later layer feature maps was performed through fixed-weight deconvolution) [14]. The maps showed network activity was generally driven by features that were salient to driving (i.e., much activity was focused on lane markings on the road, or on parked cars), but the overall interpretation was ill-defined and up to the viewer. Poor explainability is a general issue with saliency maps, and this chapter's project aims to show a low-cost example of a feature interpretation experiment that can be better functionally describe results.

CNNs for Self-Driving

Before 2018, most self-driving research in the academic domain was done on simulated driving data or broadly collated dashcam driving data. Examples of such simulated driving were usually done on environments like CARLA or GTA 5, with tools designed to record user-driven data [35, 81]. Soon to follow were other examples on dashcam data was from universities that collected their own (e.g. UC Berkeley with BDD [139]) or publicly released datasets from companies like Waymo [119] and Lyft [62]. Ultimately, the simulated environments had relatively little variability in driving situations (e.g., the same looking towns in CARLA with defined weather or lighting conditions) or were expensive to generate (e.g., the price of GTA 5 research packages). On the other end, open-source datasets for self-driving did not allow for as much control over driving situations as virtual environments.

Some research had personally collected training data for real-world evaluations, like at Carnegie Mellon University [97] or for DARPA desert driving challenges[83], but exhibited the same problems as publicly available data: data collection and training became prohibitively expensive to continue, or commonly structured environments in training and testing features did not allow results to generalize to other data. Also, the DARPA contestants did not train a CNN.

With respect to feature visualization of driving cues, there were saliency maps [14] or predictive models of self-driving distributions [135]. The former is quantitatively uninterpretable, while the latter is overly dependent on dense segmentation labels for interpretation. Motion selectivity was not well studied then, aside from action recognition tasks. Specifically focusing on optical flow evaluation, experiments at the time were also performed with dense, pixel-wise labels of image pair differences [127].

Notably unique at the time, Karl Zipser built a scaled-down remote-control car and drove it around various structured and unstructured environments using a joystick controller [61]. A CNN (SqueezeNet) was trained through imitation learning to predict output driving samples across time using video inputs. [64]. A supplementary note about all of the examples listed is that their CNNs process single image inputs with single output tasks, whereas this network processes multiple inputs to predict over a longer time period of output driving behavior (Fig 2.1).

Egomotion Cues

Self-driving is primarily an egomotion task (i.e. 3D motion of a camera within a scene). As with representation learning in general, egomotion research can be inspired by human vision experiments.

There are many non-attentional cues that can affect biological egomotion along a route. Human salience modeled after behavioral data can be driven by brightness or color contrast [66], blur and disparity [117], or other perspective cues (monocular and vestibular [20]). While most of these features are particular to a biological egomotion and may not relate to a camera-based egomotion using CNNs, some features (e.g. color, brightness, and texture) are relatable to CNNs. However, those cues are expensive to thoroughly label, and also relate to single-image vision.

Optical flow is another well-studied cue that pertains to biological egomotion [131]. It is a pattern that captures visual motion relative to an observer (e.g. eye, camera) during movement through an environment. Flow patterns generated from pure translation or pure rotation can useful for viewpoint matching and distance approximation, just like stereoscopic disparity cues. However, it differs from disparity in that a single camera or eye can experience it across time. Also, optical flow can provide cues for path centers for curved paths, so it inherently encodes future path information [107].

2.2 Methods

For this autonomous driving project hosted by Karl Zipser, my goal is to perform neuroscientific style experiments to probe artificial neuronal tunings. The task was to train a knee-high remote-controlled car top perform imitation learning. Using a joystick controller, someone would drive the car and collect training data, on which a small feedforward neural network would try to learn and reproduce driving behavior. As seen in **??**, this self-driving task would take in 10 frames of video from an RGB stereo-pair camera at a given framerate, and then would predict 10 points of steering and motor in the future at a lower frame rate (ultimately, the task is to predict across a longer time range than the inputs).

We organized input videos by their average steer and motor throttle combinations. We only used videos whose current and future driving combinations had little variation (below a certain standard deviation threshold, not reported), as well as only used videos whose future driving outputs were well predicted by the network. This allowed us to easily test on salient ego-motion videos containing one type of flow per video.

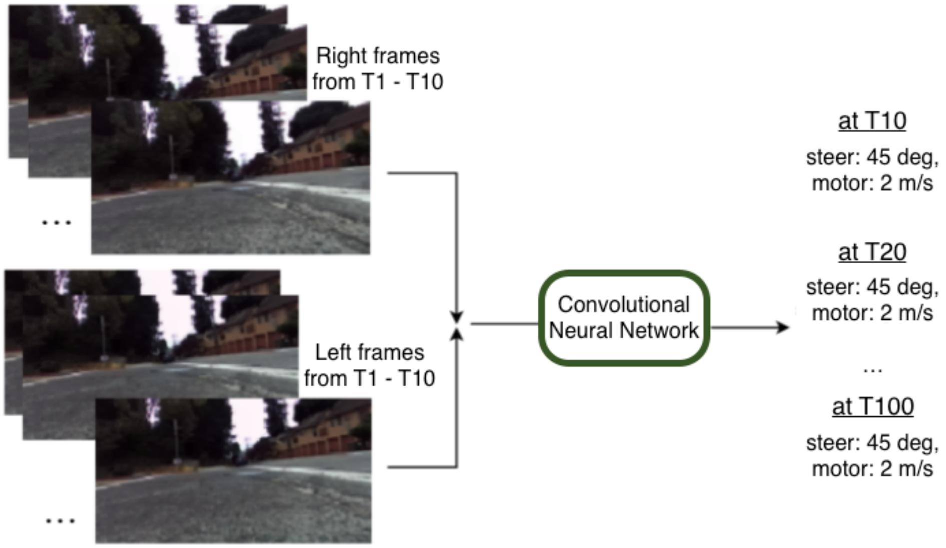


Figure 2.1: Input Task. Representative pipeline showing the self-drivinig task. Multiple input frames are fed into a network that predicts across multiple time points for both steering and motor. There is a longer time range for output predictions than input video frames with the assumption that the CNN will improve future prediction with a little bit of past data.

We first created gradient ascent style visualizations of receptive fields [11, 141]. Shown in Fig 2.2, we generated gradient ascent visualizations on Layer 1 for an shallow CNN (2 convolutional layers and 2 dense layers) taking in 2 frames at a times. Across frames and cameras for any given neuron filter, Layer 1 receptive fields appear sensitive to optical flow and natural stereoscopic disparity. We note that gradient ascent style filter visualizations directly match the filter weights for the first layer, but report the technique regardless.

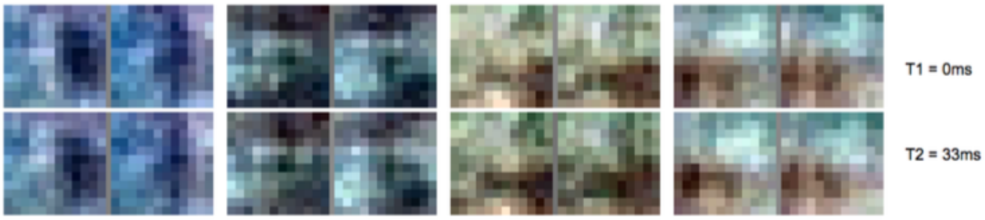


Figure 2.2: Gradient Ascent Visualizations. Shown are four neurons' receptive fields from Layer 1 of our first self-driving network. Each neuron filter is divided into sub-filters, with one sub-filter per camera, per input frame – hence the 2x2 layout per neuron filter. These filters are appear sensitive to optical flow and stereoscopic disparity.

Despite being the simplest to visualize and confirm, these filters are not easily interpretable, and later layers are even noisier, so this avenue of investigation is not feasible. Furthermore, our current convolutional network is primarily the SqueezeNet architecture from Iandola, et al. [65]. We did not want to interpret unstructured visualizations from 1x1 and 3x3 filters. Instead, though not semantic, we labeled and compared inputs by presumed relevant features, similar to Zhou, et al. [8]. We then took inspiration from the general feature manipulation of predictive modeling experiments in psychophysics [137].

We studied optical flow because they provide cues about depth and future trajectories [107], and there is early evidence for them through gradient ascent analysis.

As seen in Fig 2.3, we changed the framerate for a given video by either sampling future frames (for the speed up condition) or interpolating additional frames in between current ones (for the slow down condition). Because of our video selection process, our framerate manipulations created new videos with similar optical flow vectors across the visual field, but with more or less magnitude. In addition, the manipulated videos had approximately the same spatial frequency information from domain specific statistics.

A secondary control condition was the frame order. Before manipulating a video's frame rate, its frame order was either maintained, reversed, or randomized. This condition also evaluates the importance of frame rate manipulation relative to frame order manipulation.

We then compared how these manipulations affected output driving predictions to test the relevance of input video motion.

2.3 Results and Discussion

We assessed motion sensitivity learned by our CNN. We did not have a distribution of driving situations in each visual domain of data collected, which ranged from manicured (sidewalk)

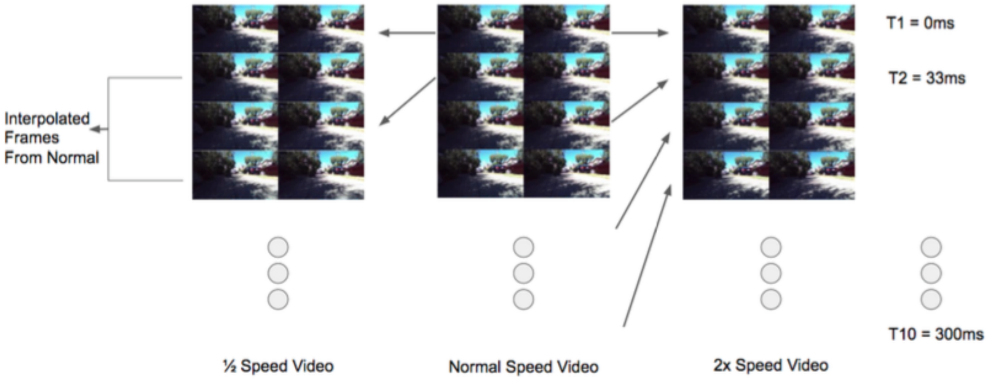


Figure 2.3: Video Speed Manipulation. Natural videos are resampled for the optical flow experiment, to simulate optical flow changes invariant of other natural features. The network expects 10-frames of input video to the network, so each manipulated video samples the original frames to match the appropriate size. Sped up versions can just use future frames, but slowed down versions need the timepoints in between the normally captured frames, which are created using the interpolation method by Meyer et al. [82]

to naturalistic (outdoors park) to in-between (on campus or tracks). Instead, we assessed network sensitivity by modifying input video samples without majorly affecting the spatial frequencies present across the video frames. Theoretically, we expected lower frame rate sampling to push predictions both steering and motor toward zero, and for higher frame rate sampling to do the opposite.

As seen in Fig 2.4, input video speed manipulation affects both steering and motor throttle predictions. This suggests potential optical flow sensitivity, but will need to be explored further.

Temporal Controls

In Fig 2.5, steer and motor throttle predictions were plotted for input videos with different frame orders. Motor throttle predictions appear robust to frame order transformations, but the steering predictions are not.

Steer Results Across Temporal Controls

As seen in Fig 2.6, changing around the frame order significantly impacts the video speed manipulation experiment for steer predictions. We need smooth flow of time, either forward or reverse, to get results similar to those from the video speed experiment in Fig 2.4. This implies optical flow filters are used for steer decisions.

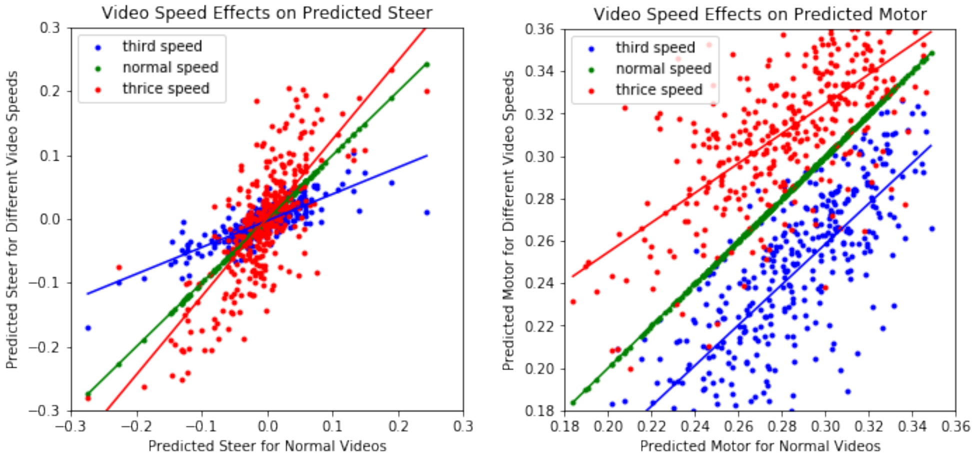


Figure 2.4: Driving Predictions After Input Video Speed Manipulation. The output steer (left) and motor throttle (right) neurons' activations with respect to video speed changes are plotted. The X coordinates are normal video predictions, and the Y coordinates are changed-speed video predictions. Zero means no behavior for both plots. The fit lines indicate that speeding up the input video pushes steer predictions to become more extreme, as well as increasing throttle predictions. The opposite is also true for slower videos.

Motor Speed Results Across Temporal Controls

For motor throttle predictions, changing around the frame order does not significantly impact the video speed manipulation experiment. Fig 2.7 shows motor throttle predictions are sensitive to input motion independent of frame order, implying that variance filters are used. Independent of frame order, little motion would yield little variance across the frames, whereas high motion would yield the opposite.

We show that our network trained to predict steering and motor throttle from stereo video exhibits different motion-selective behavior for steering and throttle. Through a series of controlled psychophysical experiments, we demonstrated that both the steer and motor throttle predictions are correctly affected by varying the motion in the input video. However, even though both behaviors look similar on the surface, correct steer predictions are dependent on smooth frame order, whereas motor throttle predictions are not.

We show that steer decisions are based on frame order in the hidden layers, whereas motor throttle decisions are not. Why are they behaving different when we expect them to act similarly? They have the same input and have the same number of output timepoints to predict, so their behavior should be identical, but they are not. This analysis shows that frame order is not salient for the underlying motor prediction task, despite the optical flow features visible in the input (as evidenced by its salience for steering prediction).

We did the same video speed experiments on hidden layer neurons as we did for the output

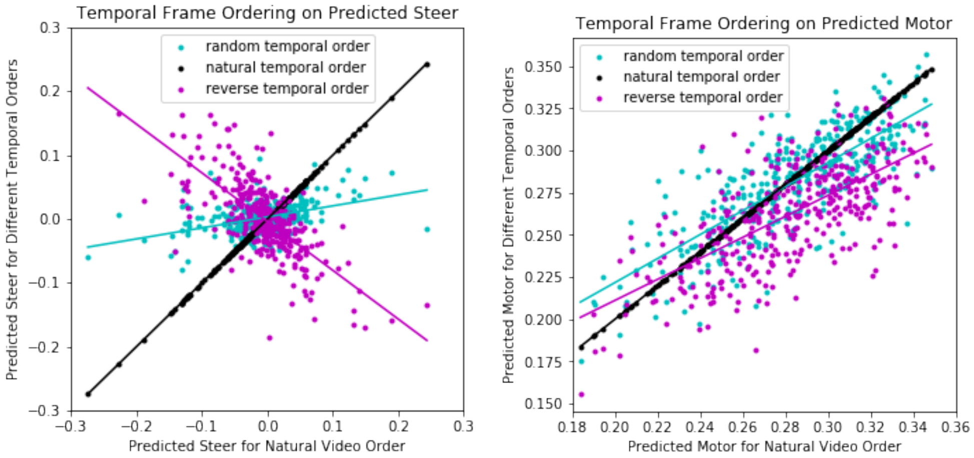


Figure 2.5: Steer and Motor Throttle Prediction Changes From Temporal Frame Ordering. Changes to output steer (left) and motor throttle (right) neurons from input frame ordering are plotted. The X coordinates are naturally ordered video predictions, and the Y coordinates are predictions after temporal ordering. The fit lines for the steer plots indicate that randomizing the frame order nullifies any steering prediction, whereas reversing the order (not in the training set) reverses the steer prediction. The fit lines for the throttle plots indicate that randomizing and reversing the frame order had little impact on the throttle prediction.

neurons. By plotting average neuron activation for changed-speed videos versus normal speed videos, we can generate the same steer-like and motor-like profiles as in Fig 2.4. For example, Fig 2.8 shows that Layer 4 has neurons that exhibit strong linear separability, just like output motor throttle neurons seen in Fig 2.4. This layer was chosen because it had the highest percentage of motor-like neurons Fig 2.10.

2.4 Conclusion

We further found the distribution of motor-like neurons across the layers (Fig 2.9), arguing that these ultimately contribute to the final steer and motor throttle predictions. Each layer clearly has neurons that exhibit motor-like tunings, and the fact that negatively-tuned neurons exist is a neat parallel to human brains [22]. Linear SVMs were used to find the motor-like neurons based on their activation profiles, with the middle layers of our network having the most motor-like neurons. By tracking the percentage of motor-like neurons in each layer, we can see that the signal dominates early filter preferences the most in Layer 4, before again being the dominant signal for output predictions (which makes sense because it is correlated with the output task). Motor-sensitivity is a feature that is learned early, and

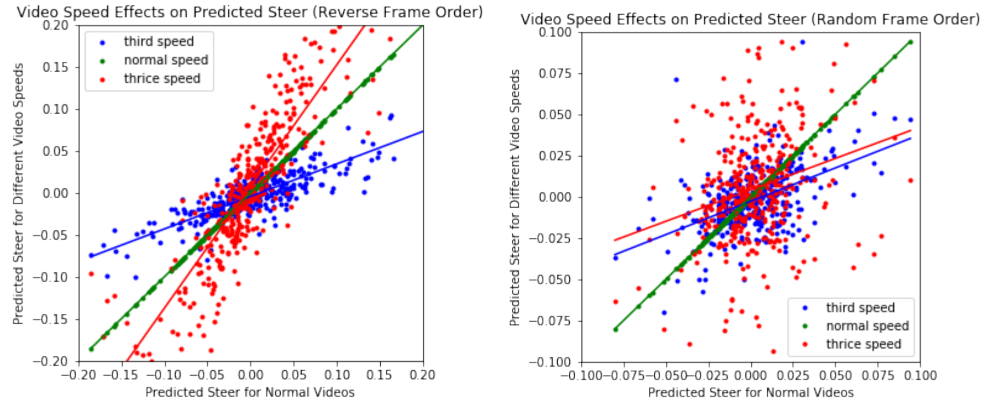


Figure 2.6: Steer Predictions Changes From Temporal Frame Ordering After Video Speed Manipulation. Here, input videos are sped up and slowed down as in Fig 2.4, but also have their frame orders changed. We can see that reversing the frame order (left) maintains the natural steer changes correlated with video speed manipulation (as in Fig. 5), but randomizing the frame order (right) breaks the natural steer prediction changes after speeding up and slowing down the videos.

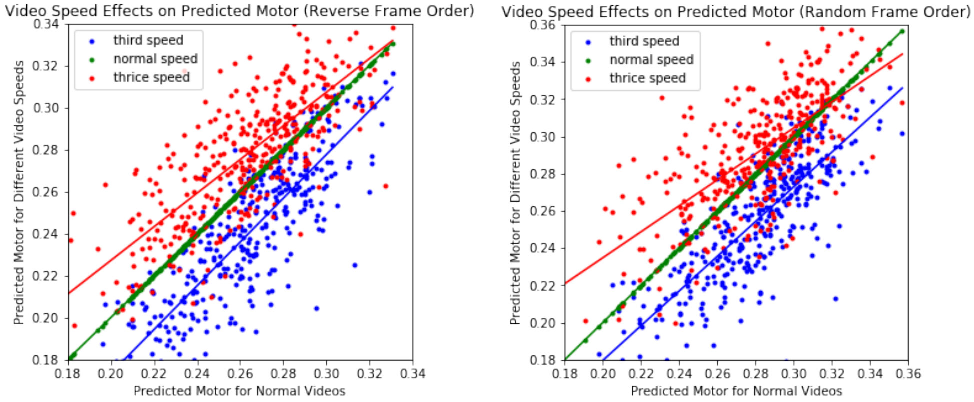


Figure 2.7: Motor Throttle Prediction Changes From Temporal Frame Ordering After Video Speed Manipulation. Here, input videos are sped up and slowed down as in Fig 2.4, but also have their frame orders changed. We can see that both randomizing the frame order (left) and reversing the frame order (right) maintains the natural throttle prediction changes after speeding up and slowing down the videos.

maintains throughout the network

This suggests a similar analysis could be performed to track steering-like neurons throughout layers of the network to identify when frame order becomes a relevant feature for repre-

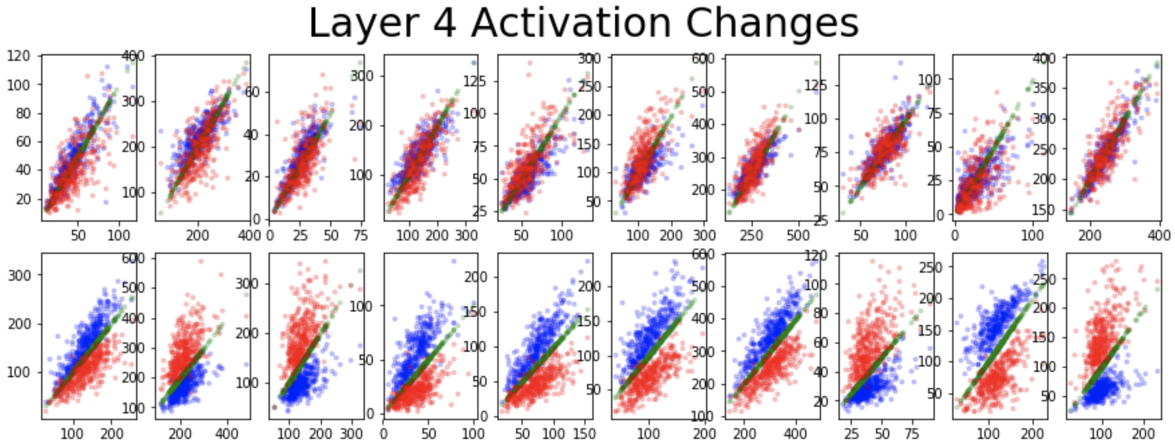


Figure 2.8: Motor Throttle Prediction Changes After Video Speed Manipulation for Layer 4. 10 Least and 10 most separable neurons, sorted by linear separability, similar to the motor throttle sensitivity profile seen in Fig 2.4.

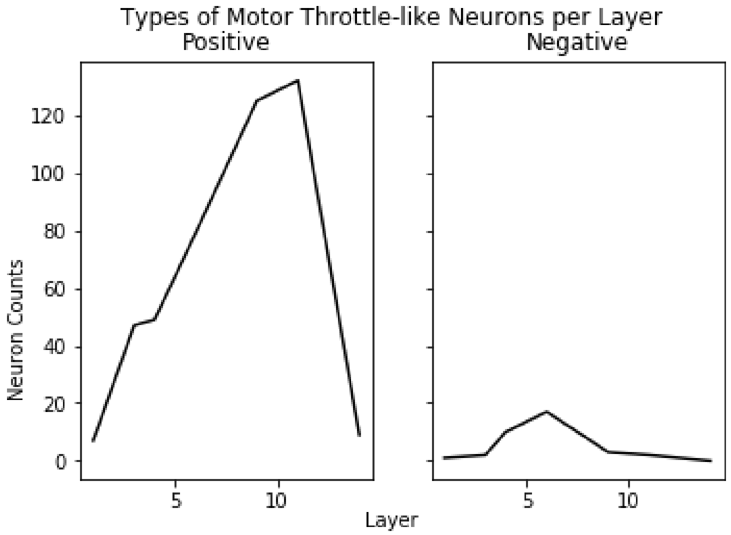


Figure 2.9: Motor Throttle Types - Positive and Negative Tunings. By tracking linear separability of neuron behavior, similar to the motor throttle sensitivity profile seen in Fig 2.4., different neurons are shown to have both positive and negative correlations to input speed changes.

sentation encoding. In addition, it is notable that steering and motor prediction were driven by the same input videos and output time points, yet exhibited different response behavior.

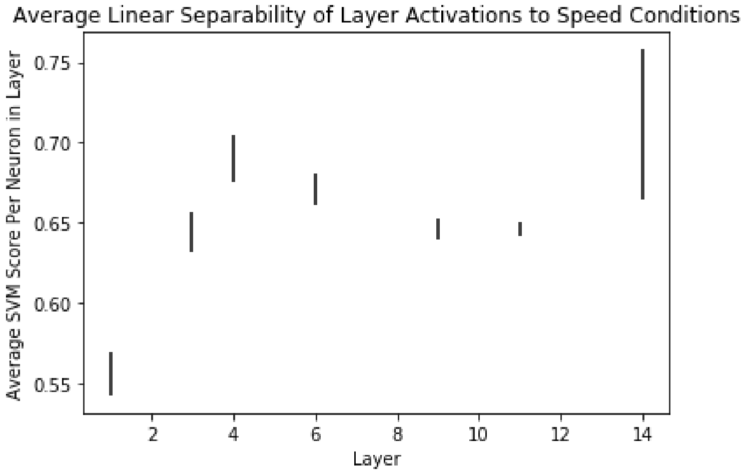


Figure 2.10: Percentage of Linear Separability Acrooss Layers. For each layer of the network, neurons are assessed for their linear separability to video speed manipulation, similar to the motor throttle sensitivity profile seen in Fig 2.4. The average separability per neuron, alongside standard error, is plotted. Motor-like sensitivities dominate layer representations the most in Layer 4.

This suggests that network behavior is specifically task relevant, and motor prediction is not causality-dependent [150], and this idea of data vs. architecture as the driver of learned features is still debated [39]. Overall, we show that neurosicence-style experiments are useful for interpreting neural network behavior, and they provide insights about the underlying task.

Chapter 3

Self-Supervised Representations for Medical Retinal Data

3.1 Introduction

Moving away from supervised learning, there are many styles of neural network that do not use expert labels to drive their high-level representation learning. Neural networks learn representations that are relevant to a task without expert labels driving the loss gradient through the labels? Given that neural networks learn similar representations despite a large variety of loss functions (aka, different tasks), it's not evident that supervision is needed for functional representation learning.

This question is always relevant because labeling is expensive, but there are applications where this cost is higher, like in medical imaging domains, so unsupervised / self-supervised learning has more inherent utility.

Self-Supervised Learning

Learning without expert labels has been of some interest since around 2015, and has approached supervised performance on some tasks since then. Some styles of techniques are exemplar learning [36, 9], image colorization [142], patch generation [92], and instance discrimination [133]. Each one performs a pre-text task, with the hope that the feature encodings can be adapted for another task. While many of these techniques learn representations that are intrinsic to image features without the guidance of expert labels, it is unclear how generalizable results are for tasks overall. For example, we would not expect an image colorizer to either colorize or attend to medical image features properly, but instance discrimination may be a more transferable prior task. As such, I will focus on instance discrimination, for the sake of generalizability.

I will specifically focus on Non-Parametric Instance Discrimination (NPID) [133]. NPID uses stored feature vectors for each training instance to help with both learning and evaluation. In general, instance discrimination assigns identifier labels (i.e. '1', '2', '3', etc.) to each image, instead of expert labels. With each pass of the data the algorithm pulls encoded features for the same image toward each other, and pushes encoded features for different images away from each other.

NPID

For learning, a given training instance's encoded output feature vector is pushed toward the feature vector stored from the previous epoch, and away from many stored negative samples from the previous epoch. NPID uses any neural network as its backbone, applies an l2-normalization to the output feature vector, and then weights the push-and-pull effect based on the distance (cosine similarity) of the normalized unit vectors. This representation learning can be boosted with learned transformation invariance. However, there is a tradeoff between better instance discrimination across negative samples and better transformation invariance across positive samples, which can be minimized with group discrimination through batch-level clustering in an updated version of the training pipeline [129].

For evaluation, a given test query instance's encoded output feature vector is compared against the nearest K neighbors of encoded training references. Then, those neighbors' true expert label is used in a weighted voted scheme, where the majority vote drives the query prediction for the task. The vote is is weighted by the distance (cosine similarity) of the normalized unit vectors between the query and training instances.

NPID was originally evaluated for ImageNet and Places datasets. These are datasets that were specifically curated for visual learning with a relatively uniform distribution of variability per class [cite], so when we look at the nearest neighbor retrievals that led to successful query predictions, it's difficult to make claims about specific features learned. However, the failure cases reveal more insights into the filters learned by instance discrimination. In the case of ImageNet failures visualized by Wu, et al., [133], the failures reveal visually similar colors, textures, and contrast between the query and retrievals of different object classes.

This nearest neighbor voting scheme allows us to investigate what specific training references drove any given query's prediction. It's a level of interpretabilty built into the algorithm itself.

Deep Learning on Medical Imaging

Deep learning is starting to replace other machine learning models in the medical domain. Applied research has been developed for CNNs with brain MRI scans [112], lung CT scans [77], eyelid images [128], and mammogram data [102]. A big disadvantage with medical data is the lack of extensively labeled data, as it takes uncommon expertise to judge images. As such, most deep learning applications have focused on image generation to augment datasets.

As an explorative study of the versatility of NPID to other tasks, we applied this technique to a dataset of images curated for medical diagnosis, not generalizable visual learning. We applied NPID to retinal fundus images (photos of the back of the eye) from the AREDS research group, as there is a growing use of deep learning for retinal classification tasks.

CNNs for Retinal Image Processing

In ophthalmology, deep learning algorithms can provide automated expert-level diagnostic tasks such as detection of diabetic retinopathy [56, 49, 123, 108, 1, 71], age-related macular degeneration (AMD) [19, 54, 93], and glaucoma [78, 30, 116] using retinal fundus images. They can also extract information including age, sex, cardiovascular risk [98], and refractive error [124] that are not discernable by human experts.

However, supervised learning approaches are trained using expert-defined labels which classify disease type or severity into discrete classes based on human-derived rubrics that are prone to bias and may not accurately reflect the underlying disease pathophysiology. Because supervised networks can only identify phenotypes that are defined by human experts, they are also limited to identifying known image biomarkers. Moreover, training labels are labor intensive to generate, typically involving multiple expert graders who are susceptible to human error. Even trained ophthalmologists do not grade retinal images consistently, with significant variability in sensitivity for detecting retinal diseases [18].

AMD

AREDS

Sponsored by the National Eye Institute, the AREDS enrolled 4757 subjects aged 55 to 80 years in a prospective, randomized, placebo-controlled clinical trial to evaluate oral antioxidants as treatment for AMD. The AREDS design and results have been previously reported [34]. The study protocol was approved by a data and safety monitoring committee and by the institutional review board (IRB) for each participating center, adhered to the tenets of the Declaration of Helsinki, and was conducted prior to the advent of the Health Insurance Portability and Accountability Act (HIPAA).The AREDS sites received informed consent from subjects, which was not necessary for this this post-hoc analysis on the fundus data; digitized AREDS color fundus photographs and study data were obtained from the National Eye Institute's Online Database of Genotypes and Phenotypes website (dbGaP accession phs000001, v3.p1.c2) after approval for authorized access, and exemption by the IRB. The median age of participants was 68, 56 precent were women, and 96 percent were Caucasian [5, 40]. Color fundus images from AREDS were previously graded by the University of Wisconsin fundus photograph reading center for anatomic features, including the size, area, and type of drusen, area of pigmentary abnormalities, area of geographic atrophy (GA), and presence of choroidal neovascularization (CNV) [5]. These gradings were used to develop a 9-step (more accurately a 9+3-step) AMD severity scale for each eye which predicts the 5-year progression risk to CNV or central GA [34], with steps 1-3 representing no AMD, 4-6 representing early AMD, 7-9 representing intermediate AMD, and 10-12 representing advanced AMD including central GA (step 10), CNV (step 11), or both (step 12) [34, 5, 40, 4] (Figure 3.1a). Both the 9+3-step scale and the simplified 4-step scale have been used to successfully train supervised CNNs to classify AREDS fundus images for AMD severity [54, 18]. As NPID's feature space is more dependent on low-level visual variety to make its prediction space less susceptible to bias, performance is bolstered by not excluding any images, such as stereoscopic duplicates or repeated subject eyes from different visits. A total of 100,848 fundus images were available, with a long-tailed imbalance and overrepresentation of the no-AMD classes for both scales, and class 11 (CNV) in the 9+3-step scale (Figure 3.1b-c). Images were randomly partitioned into training, validation, and testing datasets in a 70:15:15 ratio, respectively, while ensuring that fundus images from the same subject did not appear across different datasets.

The method used for labeling requires medical expertise, so it is more expensive than ImageNet or Places annotations. Fig 3.2 shows that visual defect areas (i.e. drusen, GA, CNV) have to be estimated according to different sizes (e.g. C-0, I-2, 0.5 DA) and then according to different eccentricities (the concentric circles). This lends to the question of, is this expertise needed for visual learning for AMD sevrity classification?

3.2 Methods

We pre-processed images according to a standard procedure that has worked for supervised visual learning in this domain. This pre-processing significantly improves performance by over 15 percent for the coarse task (not shown).

Fundus images were down-sampled to 224x224 pixels along the short edge while maintaining the aspect ratio as similarly done in past literature [54]. Fundus images were also preprocessed with a Laplacian filter applied in each of the red-green-blue (RGB) color dimensions to better emulate the properties of more natural images of everyday scenes and objects (Figure 3.3). Laplacian filtering is the difference of two Gaussian-filtered versions of the original image. In this study, it is the original fundus image (effectively, a Gaussian-filtered image with no blur) subtracted by the image Gaussian-filtered with a standard deviation (SD) of 9 pixels in each of the RGB color channels.

As a minor point, neural networks (like ResNet) and NPID were originally developed and evaluated on images with natural statistics. A tenet of natural statistics is the 1/f distribution of spatial frequencies of a given image, where lower frequencies are more represented than higher frequencies [105]. Fundus photographs exhibit approximately the 1/f power distribution of natural images of everyday scenes and objects 33,34 but with more low-frequency than high-frequency information (Figure 3.3a). The Laplacian-filtered fundus images more closely resembles that of natural statistics (Figure 3.3b).

Network Pretraining

A CNN can transfer knowledge from one image dataset to another by using the same or similar filters [149]. Unlike natural images that contain a variety of shapes and colors that are spatially distributed throughout the image, fundus photographs are limited by shared fundus features such as the optic disc and retinal vessels, as well as the restricted colors of the retina and retinal lesions. This in turn limits the variability of the filters learned by the network. Thus, to transfer learning from a higher variety of discriminable features, we pretrained the network using the large visual database ImageNet (i.e., initialize the neurons across naturalistic filters), and then finetuned on the AREDS dataset without any weights frozen to further improve performance. A comparison of different sizes for the final layer feature vector for NPID, which depends on the complexity of the filters learned from the task, revealed an ideal size of 64 dimensions for our pretrained model to maximize the performance gained from transfer learning (Figure 3.4a).

As seen in 3.5, we use the same general pipeline for training and evaluation as had been originally done for ImageNet and Places.

For wkNN, we chose k=12, as it produced the highest balanced accuracies (Figure 3.4b). We chose the epoch that yielded the best balanced accuracy using wkNN classification voting scheme (see Appendix A). Then, we evaluated that epoch on a separate testing dataset using various metrics from the wkNN result including unbalanced accuracy, Cohen's kappa, true positive rate, and false positive rate. Unbalanced accuracy is the average accuracy across all samples, whereas balanced accuracy is the average class accuracy 36,37. While both accuracy metrics are relevant and positively highlight the performance of NPID, balanced accuracy is less biased to skewed class distributions by weighting underrepresented class scores as equally as overrepresented ones, and is more appropriate for comparing performance across different subsets of the same data as in our study. We also employed a second method to evaluate self-supervised features using Linear Support Vector Machines (Linear SVMs) 35.

Supervised Training & Prediction

To establish our own baseline, we perform supervised finetuning on ResNet-50 with the 9+3-step severity scale, after pretraining on ImageNet, using the same set of AREDS fundus photographs. The data augmentations and hyperparameters match that of our best implementation of NPID. To avoid retraining for each new scale, we mapped the logits from the 9+3-step scale to 4-step, 2-step advanced AMD, and 2-step referrable AMD classes to generalize coarse-grained performance. This baseline network is established to evaluate how our NPID-trained representations from fundus images without expert labels compare to those from a network supervised-trained with expert labels.

t-SNE visualization & Search Similarity

To assess neighborhoods of learned features, we evaluated search similarity and t-Distributed Stochastic Neighbor Embedding (t-SNE) visualizations. Search similarities show how a given query image's severity is predicted based on nearest neighbor references, and t-SNE visualizations show us how all the fundus images are distributed across neighborhoods of visual features chosen by the network. Specifically, t-SNE maps feature vectors from high-dimensional to low-dimensional coordinates while approximately preserving local topology. Here, we map the encoded 64D features onto 2D coordinates, wherein coordinates that are near each other in 2D are also near each other in the original feature space, meaning they are similarly encoded because they share visual features. Although t-SNE visualizations can distort some mapping from high dimensional to 2D feature spaces, our claims about NPID feature groupings were confirmed by visual review by a board-certified ophthalmologist (GY), and are thus based on the original images. The t-SNE visualization is used as a tool to discover these images faster for additional review. Thus, we can color each 2D coordinate by the known labels for each fundus image in the training set to observe which images are encoded near to each other and what visual groupings emerge from these locally similar encodings. This process is label agnostic, so evaluation across multiple domains of labels (e.g. 2-step AMD severity, 4-step AMD severity, drusen count, media opacity, etc.) is possible without retraining, unlike a supervised-trained network.

Hierarchical Learning

Because NPID appears more suitable for coarse-level than fine-level classification across dependent classes, we split up the 9+3-step dataset into each of the 4-step classes. We trained the NPID network on only no, early, intermediate, or advanced AMD images, then evaluated NPID's ability to discriminate between the three fine 9+3-step classes within each coarse 4-step class to identify which of the 9+3-step classes appear to show less visual discriminability than the grading rubric suggests.

3.3 Results

Accuracy in grading AMD severity We first evaluated NPID performance on a 2-step discrimination task for detecting advanced AMD (CNV and/or central GA), and found that our self-supervised-trained network achieved an unbalanced accuracy (94%) that is comparable to the performance of our supervised-trained CNN (95.8%), a similar published supervised network (96.7%) or trained ophthalmologist (97.3%) [93]. The balanced accuracy, which is more applicable due to dataset imbalance, was also similar between the self-supervised-trained NPID (82%), our supervised-trained network (92%), the published supervised network (81%), and ophthalmologist (8%) (Figure 3.6a). Next, we compared the balanced accuracy of NPID with another supervised algorithm to distinguish "referable" AMD (intermediate or advanced) from no or early AMD, and found that our self-supervised-trained network performed only slightly worse (87%) than our supervised-trained network (90%), the published supervised network (92%), and ophthalmologist (96%) [18], despite never learning the class definitions directly (Figure 3.6b). For grading AMD severity using the 4-step scale, NPID achieved a 65% balanced accuracy, which was comparable to our supervised-trained network (75%), the published network (63%), and ophthalmologist (67%)(Figure 3.6c) [18]. In particular, the confusion matrix for NPID demonstrated superior performance for distinguishing early AMD (class 2) as compared to both the published supervised network and human expert (Figure 3.6d) [18].

When applied to a finer classification task, NPID only achieved a balanced accuracy of 25% on the 9+3-step scale, as compared to 40% using our supervised-trained network and 74% using the published supervised network [54] that utilized the same backbone network as our NPID approach. We achieved this balanced accuracy score using k=12 for wkNN, although we also tested k=5, 8, 23, and 50, and found that results were mostly consistent across different k-values (Figure 3.4b). Even though our most class-homogenous neighborhoods are defined by k=12 neighbors, they are still mostly coherent with k=50 neighbors, which was how NPID was originally evaluated on the ImageNet dataset 24. With k=50, 28% shared the query image's label while 68% were within 2 steps of the correct 9+3-step label (Figure 3.6e). Even for cases with incorrect 9+3-step class predictions, the 50 nearest neighbor images shared the query's 4-step class label 56% of the time, which accounts for the higher accuracy of our network in the 4-step classification task. Thus, although self-supervised learning achieves lower supervised wkNN performance on the finer 9+3-step AMD severity scale compared to binary or 4-step AMD classifications, incorrect predictions deviate minimally from ground-truth labels. We confirmed our findings using linear SVM classifiers, which achieved a 26% balanced accuracy for 9+3-step classification consistent with the wkNN results.

We can see that functionally, instance discrimination learns representations that are salient to coarse-level classification. Matching human performance, especially for the the class hardest to re-grade by humans, implies our implementation is maximizing performance on this dataset.

Similar to learning on ImageNet, the nearest neighbor retrievals from NPID learning show successful classification is driven by features that are correlated with the task (i.e. object features for ImageNet classes, disease phenotypes for 12-step AMD severity classes). However, the same could be said for many failed classifications.

The apparent visual similarity between query and retrievals for failure cases leads to two questions: (1) Is the 12-step scale necessary for effective disease classification, especially when the primary use of this standard is to estimate closeness to advanced AMD, and (2) Is NPID organizing features that are relevant to other clinical applications, as well?

Network Behavior

To discern how the NPID network visually organizes images from different AMD classes, we employed t-SNE visualizations which mapped encoded 64-dimensional features onto 2-D coordinates. On the 4-step AMD severity scale, fundus images with no (blue), intermediate (yellow), and advanced (red) AMD formed distinct clusters, while early AMD (aqua / green) images are scattered throughout the plot (Figure 3.7a), which likely explains the lower performance in this class (Figure 3.6d). On the 9-step AMD severity scale (Figure 3.7b), the t-SNE plot appear similar to that of the 4-step scale, as each of the 4 major classes on the simplified scale are dominated by one or two of the finer classes within each subset (Figure 3.1b), and may account for the poorer performance of our self-supervised-trained network on the 9+3-step task.

To determine which AMD features contributed most to the self-supervised learning, we mapped AREDS reading center-designated labels including (1) drusen size, area, and type, (2) depigmentation or hyperpigmentation area, and (3) total or central GA area onto the t-SNE plots (Figure 3.8). We found that drusen area provided the most visually distinct clusters that matched the separation of the 4-step severity scale. GA area and depigmentation correlated well with advanced AMD classes as expected, while larger drusen size or soft drusen type corresponded to intermediate AMD classes.

We can do more than color-code instances by ground truth severity, like color-code them according to K-means clustering performed on the original 64D vectors (K=4). This shows that there is a clear representational boundary between instances with and without AMD severity. If we investigate individual neighborhoods along this boundary, and separate each neighborhoods by its K-means clusters (so the separation is in the original feature space), we see neighborhoods correlated with other domains.

To identify other physiologic or pathologic phenotypes beyond AMD features, we performed K-means clustering on all training images using a K-value of 4, based on the presence of 4 coarse classes in the 4-step severity scale. We observed one cluster (Cluster A) which correspond to images with no AMD, and three other clusters (Clusters B, C, and D) which appear to straddle AMD classes, suggesting that these latter groups may be distinguished by features unrelated to AMD pathophysiology (Figures 3.10a-b). A closer examination of cluster B images near the border between AMD and non-AMD classes revealed eyes with a prominent choroidal pattern known as a tessellated or tigroid fundus appearance (Figure 3.10c) – a feature associated with choroidal thinning and high myopia 41. Cluster C images near this border contain fundi with a blonde appearance (Figure 3.10d), often found in patients with light-colored skin and eyes, or in patients with ocular or oculocutaneous albinism 42. Images from cluster D in this area showed poorly-defined fundus appearances that were suspicious for media opacity (Figure 3.10e). To determine if this cluster may include eyes with greater degrees of lens opacity, we overlaid the main t-SNE plot with labels for nuclear sclerosis, cortical cataracts, or posterior sub-capsular opacity from corresponding slit lamp images obtained in AREDS, and found that eyes in cluster D corresponded to a higher degree of both nuclear and cortical cataracts (Figure 3.9). Hence, fundus images contain other ophthalmologically-relevant information that are not constrained to the retina, and K-means clustering of retinal images can also identify eyes with tessellated or blonde fundi as well as visually-significant cataracts.

3.4 Conclusion

In this study, we successfully trained a self-supervised neural network using fundus photographs which could be used with a supervised evaluation method to predict AMD severity across different human-defined classification schema, reveal AMD features that drove network behavior, and identify novel pathologic and physiologic ocular phenotypes, all without the bias and constraints of human-assigned labels during the training process. NPID performance was comparable to a supervised-trained CNN using the same backbone network, previously-published supervised networks, and human experts in grading AMD severity on a 4-step scale (none, early, intermediate, and advanced AMD) [18], and in binary classification of advanced AMD (CNV or central GA) [93] and referable AMD (intermediate or advanced AMD) [18]. Our self-supervised-trained network also performed similarly to a supervised-trained network that was trained with both fundus images and genotype data on a custom 3-step classification of class 1, class 2-8, and class 9-12 on the 9+3-step severity scale (65% vs. 56-60%) [136]. Our results suggest that even without human-generated labels during training, self-supervised learning with parameter-frozen supervised evaluation can achieve predictive performance similar to expert human and supervised-trained neural networks.

Self-supervised learning using NPID has significant advantages over supervised learning. First, eliminating the need for labor-intensive annotation of training data vastly enhances scalability and removes human error or biases. Also, NPID predictions resemble ophthalmologists more closely than do supervised networks (Figure 3.6d). Like humans, the self-supervised-trained NPID network considers the AMD severity scale as a continuum and the relationship of adjacent classes. By contrast, supervised-trained algorithms generally assume independence across classes, are susceptible to noisy or mislabeled images, and may produce more egregious misclassifications. Because the NPID algorithm groups images by visual similarity rather than class labels, inaccurate predictions can be salvaged by other nearest neighbors during group voting.

In our study, we probed the NPID network's behavior and found that AMD features such as drusen area drove predictions of AMD severity more than drusen size or type, or area of pigmentary changes. Using hierarchical learning and spherical K-means clustering, we also identified eyes with non-central GA among those with intermediate or advanced AMD based on proximity to eyes with central GA (class 10), even though this feature is not encoded in the human-labeled AMD severity scales. Our findings suggest that self-supervised learning can more objectively identify certain AMD phenotypes such as drusen area or GA presence which may better reflect disease pathophysiology, and enable the development of more unbiased, data-driven classification of AMD severity or subtypes that could better predict disease outcomes than human-assigned grades. Interestingly, K-means clustering also identified images with central GA that appeared mislabeled as intermediate AMD, further highlighting the ability of an self-supervised-trained network to discover miscategorized images in ways that label-driven supervised learning cannot.

Another notable feature of self-supervised learning is the ability to identify non-retinal phenotypes from fundus images, including camera artifacts (lens dirt or flare), media opacity (cataracts or asteroid hyalosis), and choroidal patterns (tessellated or blonde fundus). While we identified these features by spherical K-means clustering using a K-value of 4, additional cluster resolution could unveil additional pathologic or physiologic phenotypes. Future studies using von-Mises mixture models for spherical K-means clustering, which do not assume identical cluster size, may enable smaller, localized clusters of phenotypic groupings to be identified. Thus, the application of NPID may not be limited to AMD grading, and its potential supersedes that of supervised-trained networks that are limited to the classification task for which it is trained.

The takeaway from this project is that we should shift our mentality of how to tackle medical datasets. The general procedure is to label all collected images, but that requires expensive expertise and leads to bias that can be irrelevant to disease progression. In addition, cluster analysis reveals that medical images like retinal images can represent features across different tasks. This suggests tools like NPID could assist clinicians organize and label medical images distributed across multiple tasks with minimal bias and learned data-driven groupings.

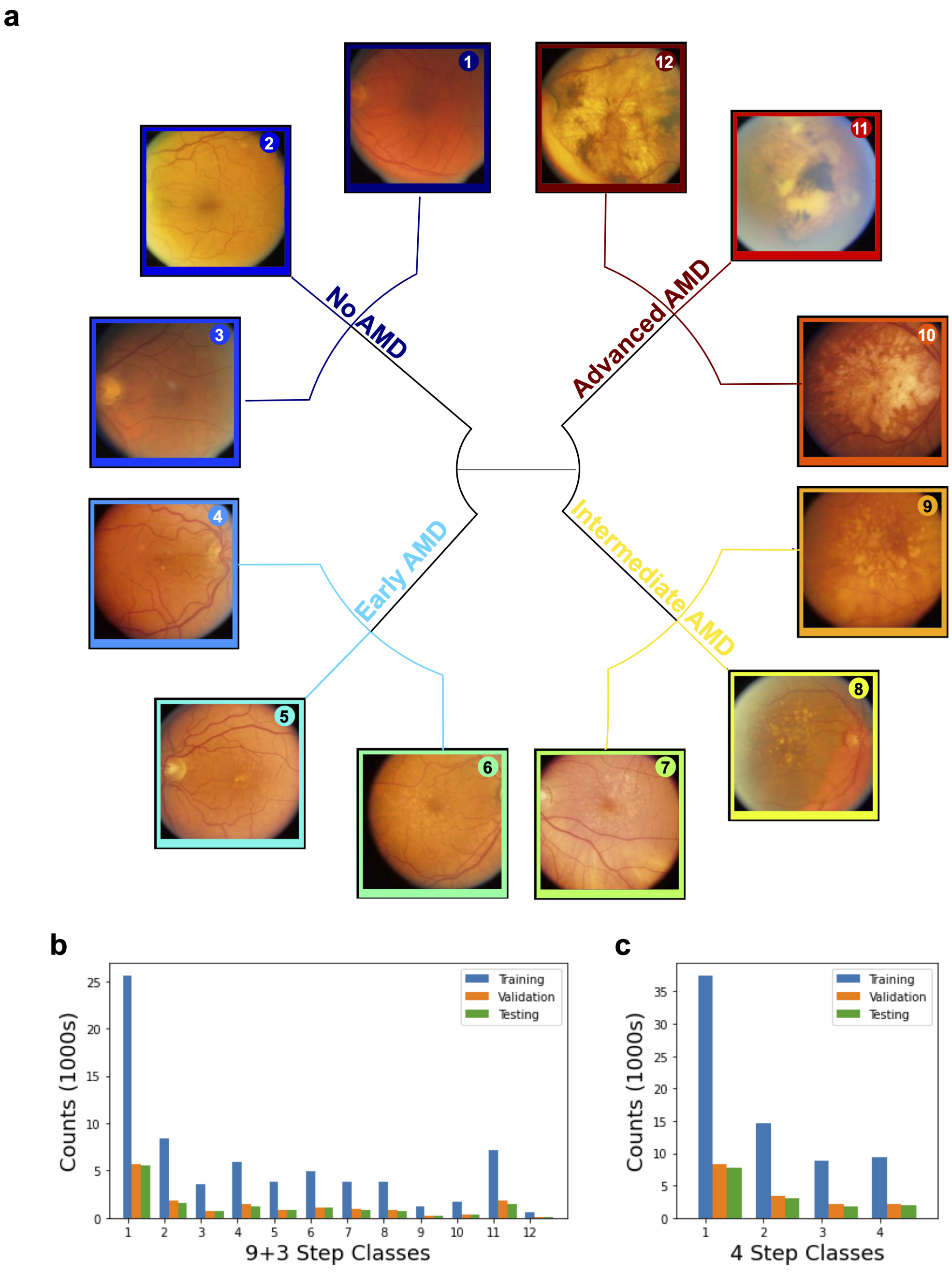


Figure 3.1: 9+3-step and 4-step AMD severity scales & data distribution. (a) Dendogram showing representative images from each of the 9+3-step AMD severity classes as defined by the reading center for AREDS, and corresponding simplified 4-step AMD severity classes including no AMD (blue), early AMD (aqua), intermediate AMD (yellow), and advanced AMD (red). (b-c) Histogram plots across training, validation, and testing labels for the (c) 9+3-step and (c) 4-step AMD severity scales, across a random 70:15:15 split of the dataset.

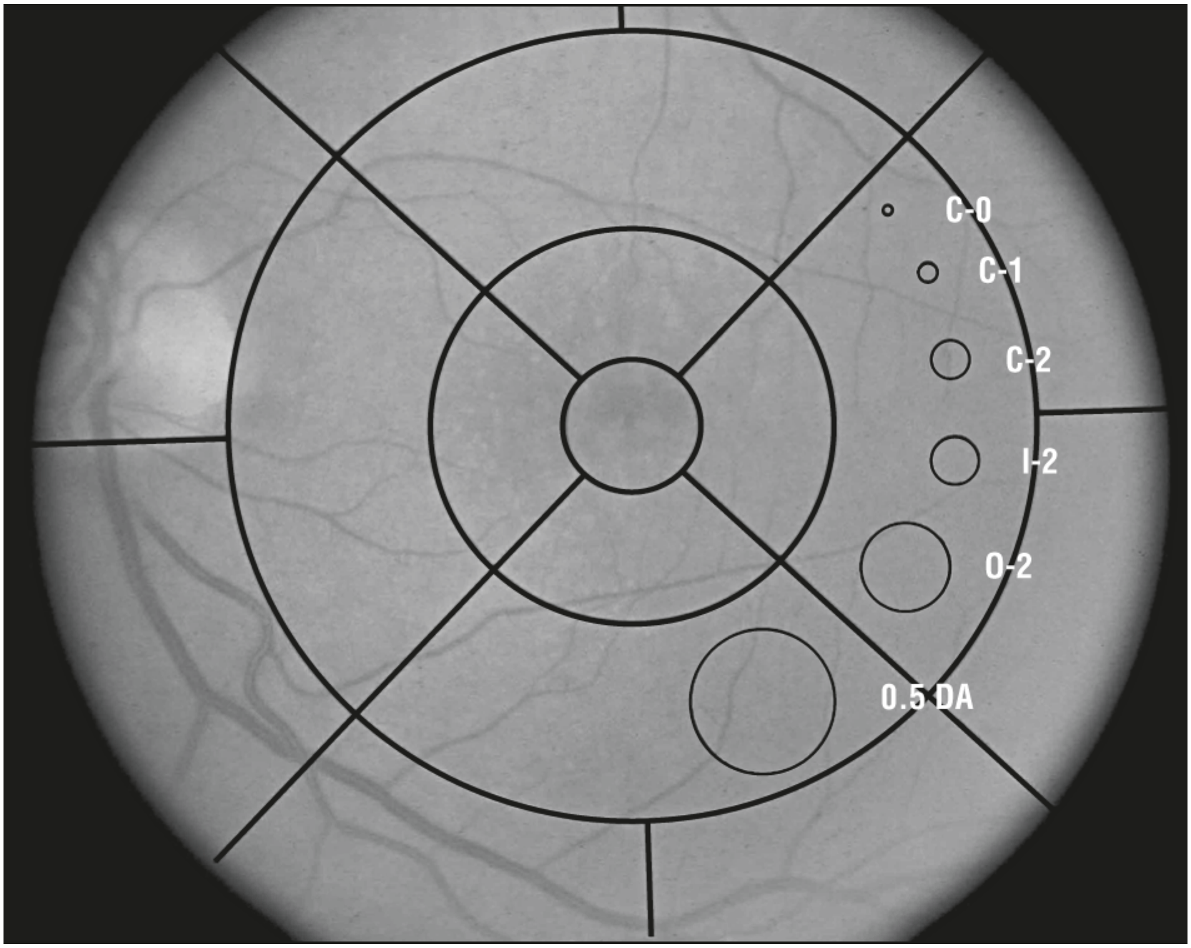


Figure 3.2: AREDS-defined grading rubric for AMD feature detection. Grid circles are at 1/3, 1, and 2 disc diameters with a standard inner circle diameter of 500 μm; The standard circles have the following diameters and areas: C-0, 63 μm and 0.0017 DA; C-1, 125 μm and 0.0069 DA; C-2, 250 μm and 0.028 DA; I-2, 354 μm and 0.056 DA; O-2, 650 μm and 0.19 DA; and 0.5 DA, 1061 μm and 0.50 DA. More detail can be referenced in Figure 3.11

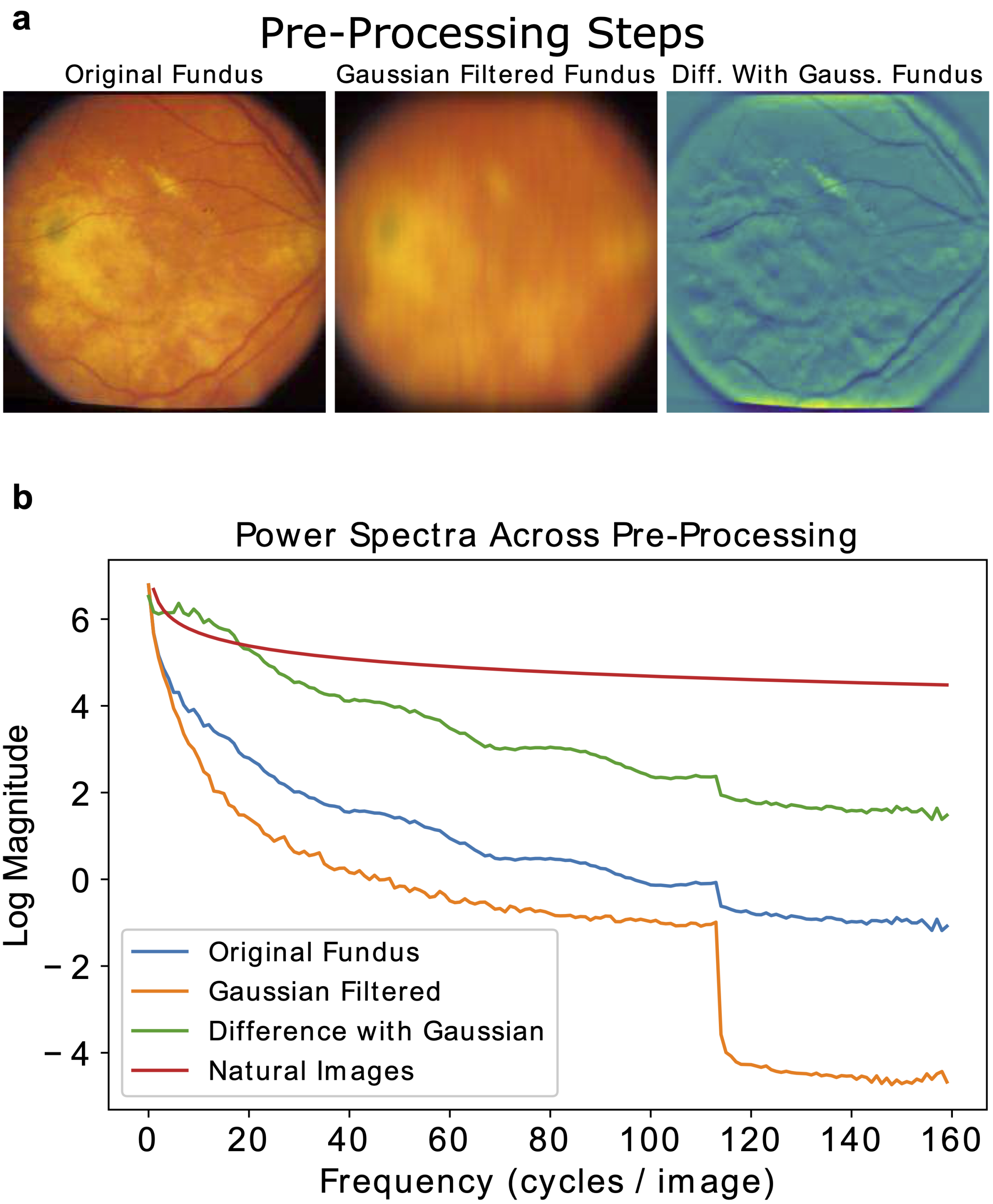


Figure 3.3: Preprocessing steps for difference with Gaussian filtering.(a) Comparison of preprocessing steps on representative fundus image, and (b) corresponding azimuthally-defined 1D power spectrum. Blue, orange, and green power spectra lines correspond to the images in (a), while the red line corresponds to the power spectrum of natural images.

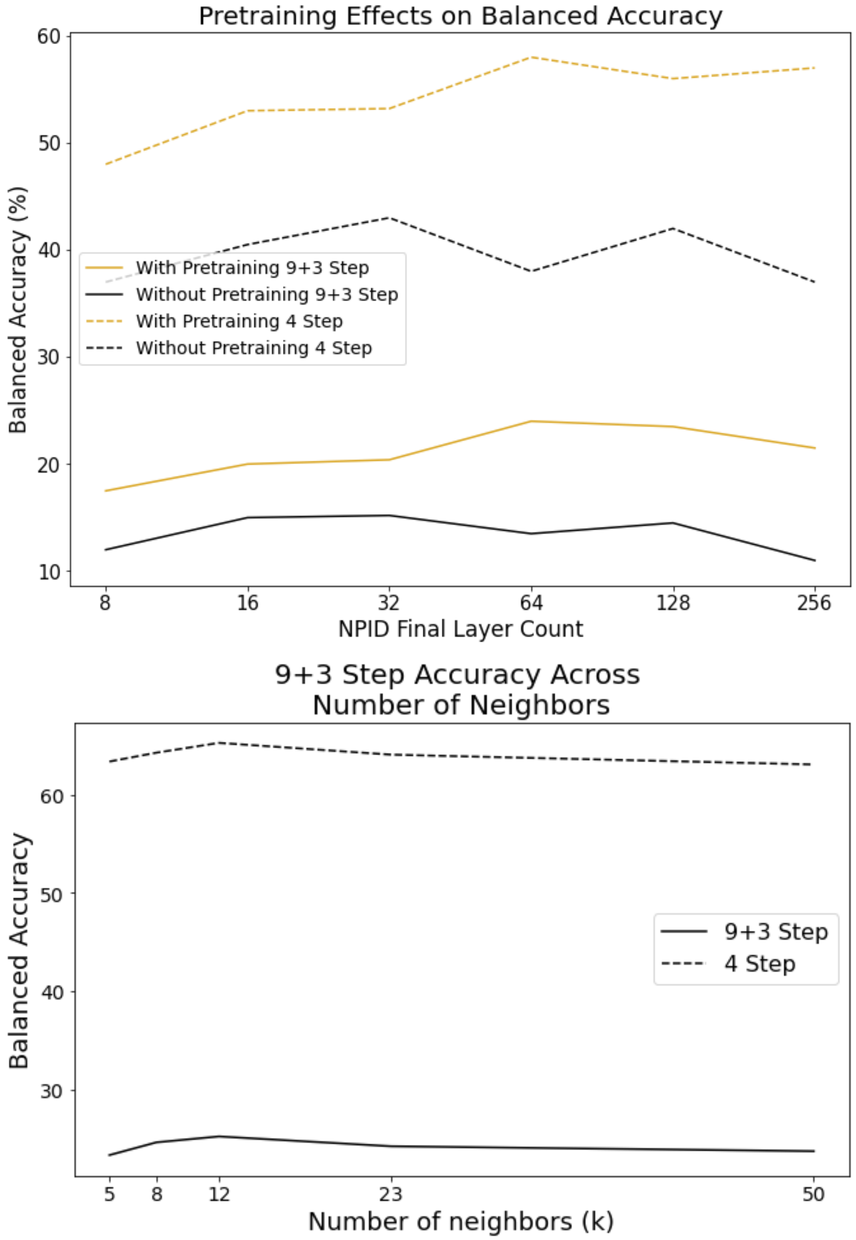


Figure 3.4: Balanced Accuracy Across Different Final Layer Sizes for NPID (Top) and Different K for wkNN voting (Bottom). (a) Representative plot showing the balanced accuracy for NPID trained at different sizes for the final layer representations, with and without transfer learning from ImageNet (i.e. pretraining). Solid lines correspond to the AMD severy classification task for the 9+3 Step labels from the AREDS Reading Center and dashed lines correspond to the 4 Step labels from the AREDS Reading Center. "With Pretraining" means the network was first pretraind on ImageNet with NPID and then finetuned on AREDS data, whereas "Without Pretraining" means only training on AREDS. (b) A plot showing the balanced accuracy results for the predictions determined from weighted k-Nearest Neighbors, across different values of k, on the output vectors derived from ResNet-50 trained using NPID.

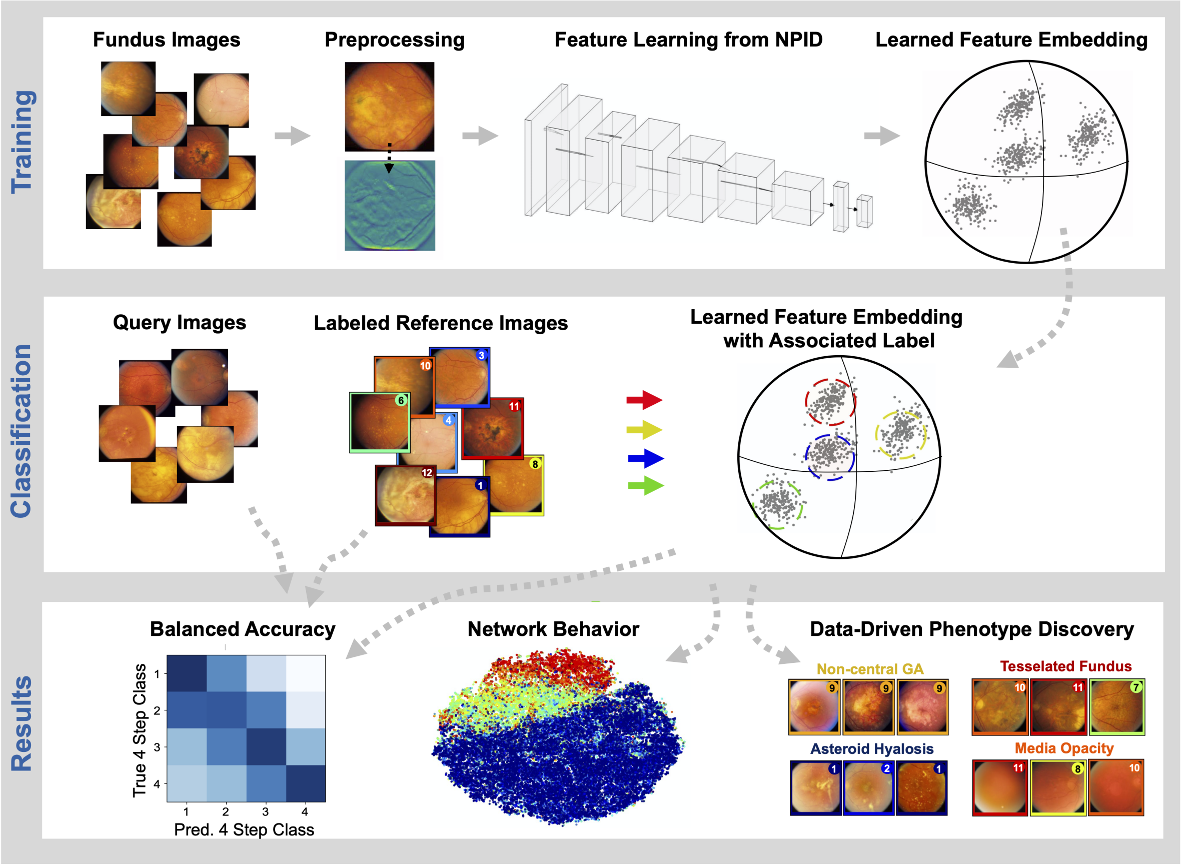


Figure 3.5: Schematic of NPID training & testing on retinal fundus images. Schematic diagram of the process by which Non-Parametric Instance Discrimination (NPID) trains a self-supervised neural network to map preprocessed fundus images to embedded feature vectors. The feature vectors and associated AMD labels are used as a reference for queried severity discovery through neighborhood similarity matching. The NPID network can then be analyzed to measure balanced accuracy in AMD severity grading, explore visual features that drive network behavior, and discover novel AMD-related features and other ocular phenotypes in a data-driven manner with minimal bias.

a Advanced AMD (2-step)

	Unsupervised NPID	Supervised ResNet-50	Published Classifier*	Human Expert*
Unbalanced Accuracy	94%	95.8%	96.7%	97.3%
Balanced Accuracy	82%	92%	81%	89%
True-positive Rate (TPR)	65%	88%	63%	80%
False-positive Rate (FPR)	1.1%	2.8%	1.3%	1.7%
Kappa	.6	.83	.66	.75

b Referable AMD (2-step)

	Unsupervised NPID	Supervised ResNet-50	Published Classifier#	Human Expert#
Unbalanced Accuracy	89%	92%	93%	95%
Balanced Accuracy	87%	90%	92%	96%
True-positive Rate (TPR)	82%	84%	95%	95%
False-positive Rate (FPR)	7.9%	5.1%	8.9%	3.6%
Kappa	.75	.79	.85	.89

c AMD Severity (4-step)

	Unsupervised NPID	Supervised ResNet-50	Published Classifier#	Human Expert#
Unbalanced Accuracy	70%	80%	79%	76%
Balanced Accuracy	65%	75%	63%	67%
True-positive Rate (TPR)				
False-positive Rate (FPR)				
Kappa	.64	.70	.70	.66

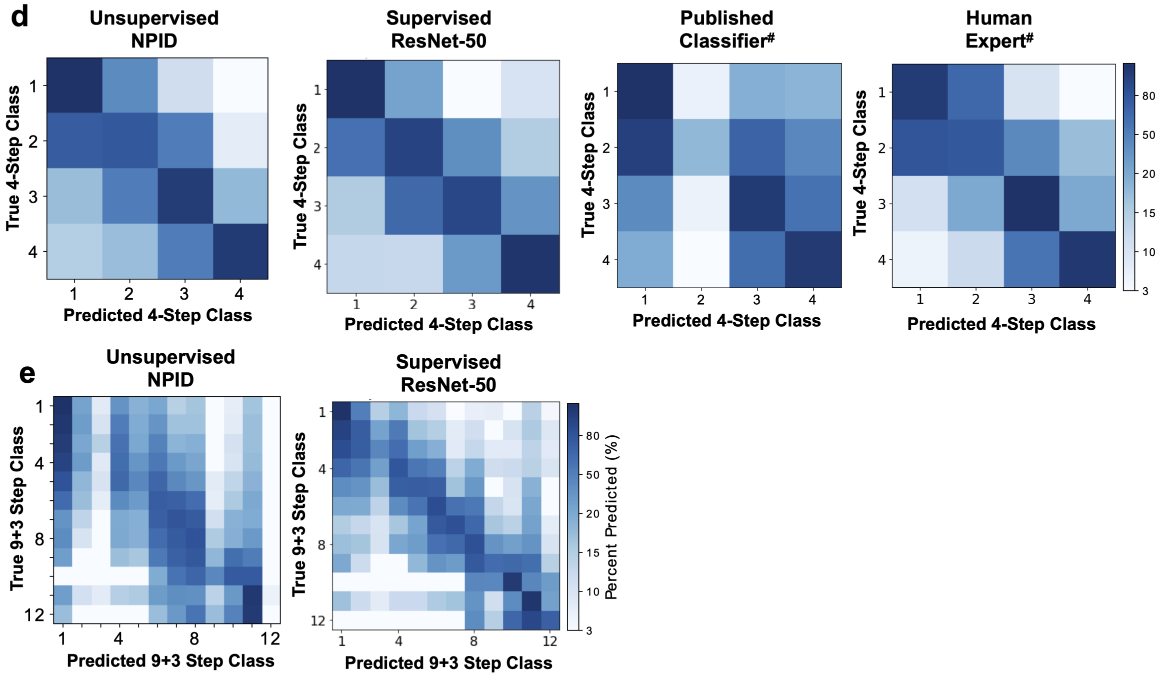


Figure 3.6: Comparison of NPID-trained performance with supervised-trained networks and human experts. (a-c) Comparisons of the self-supervised-trained NPID network performance with a supervised-trained ResNet-50 network, as well as published supervised baselines and human ophthalmologists as reported by *Peng, et al. [93] and #Burlina, et al. [18] for binary classification of advanced AMD (a) or referable AMD (b), as well as the 4-step AMD severity scale (c). (d) Comparison of confusion matrices of our self-supervised-trained network with our supervised-trained network, published supervised baselines, and human expert gradings reported in #Burlina, et al. [18] for the 4-step AMD severity scale task. (e) Confusion matrices of the NPID network and our supervised-trained network on the 9+3-step AMD severity classification task.

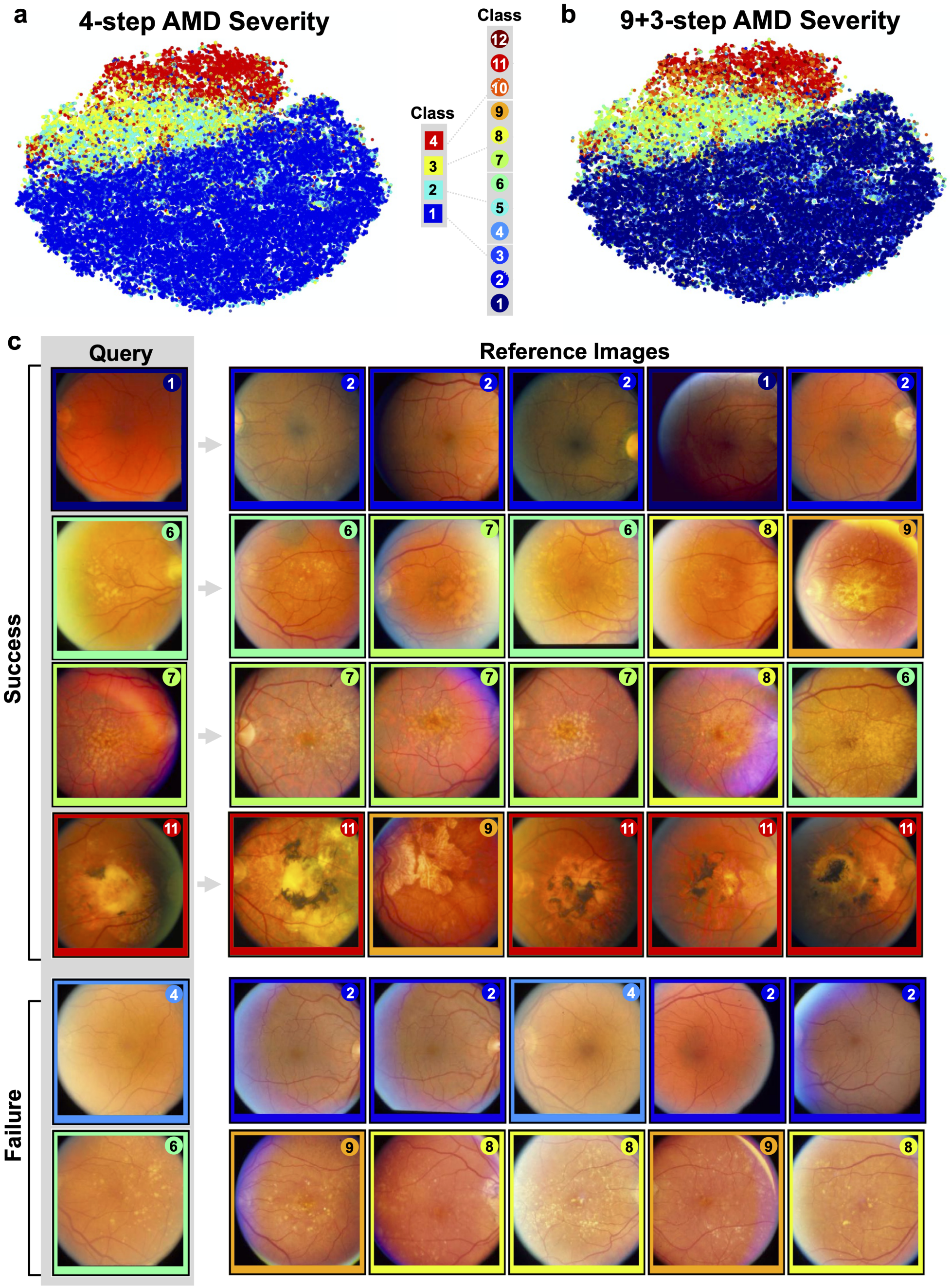


Figure 3.7: Self-supervised NPID clusters fundus images based on visual similarity. t-Distributed Stochastic Neighbor Embedding (t-SNE) visualizations of NPID feature vectors colored by (a) 4-step and (b) 9+3-step AMD severity labels, where each colored spot represents a single fundus image with AMD severity class as described in the legend of Figure 3.1. (c) Representative search similarity images for successful and failed cases for the 9+3-step AMD severity scale task. The leftmost column corresponds to the query fundus image, while the next 5 images on each row correspond to the top 5 neighbors as defined by network features. The colored borders and numeric labels for each image define the true class label defined by the reading center for AREDS, and correspond to the color scheme in Figure 3.1

.

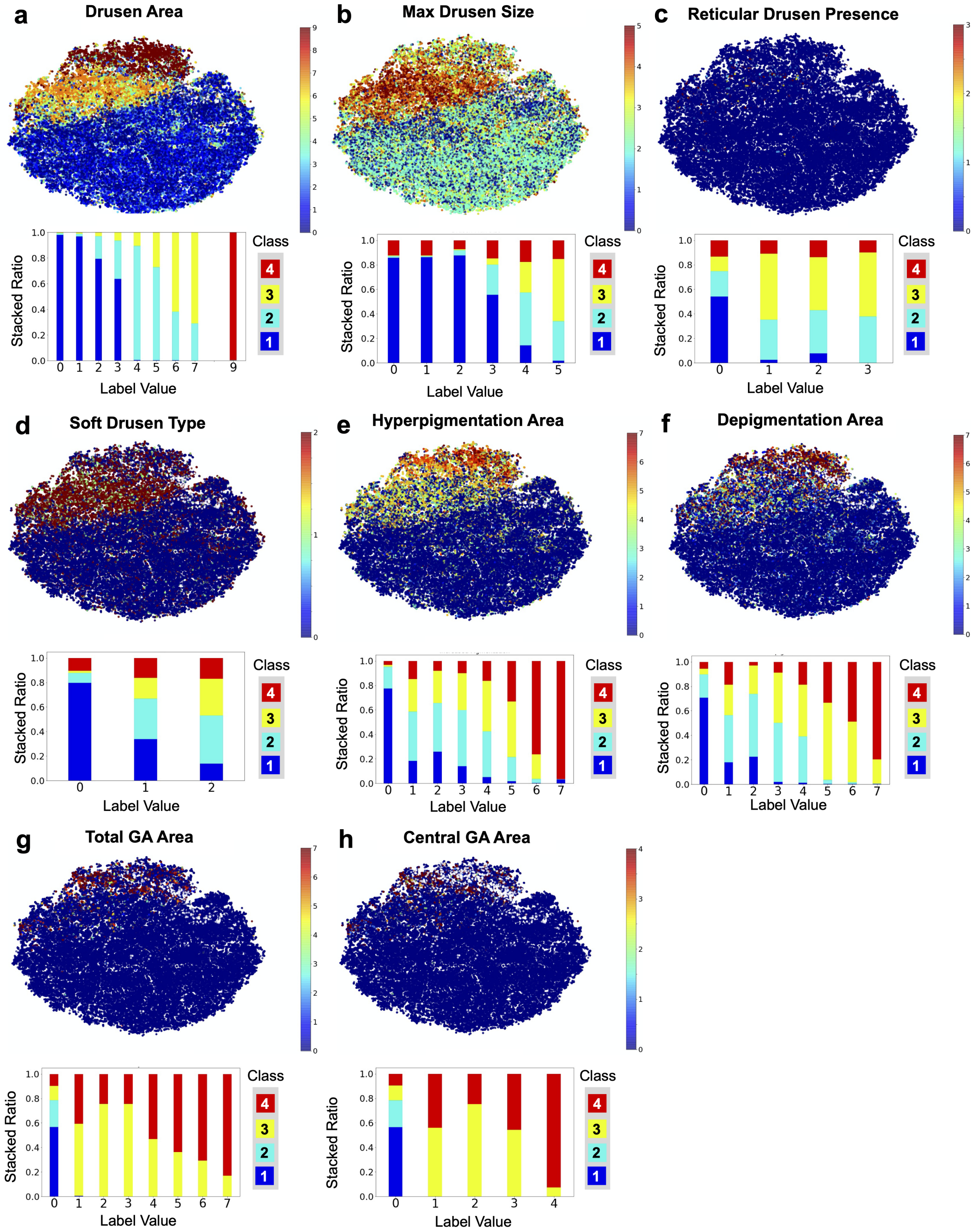


Figure 3.8: AMD-related fundus features that drive NPID-trained network predictions. t-Distributed Stochastic Neighbor Embedding (t-SNE) visualizations of NPID feature vectors colored by AREDS reading center labels for AMD-related fundus features, with corresponding stacked bar plots showing ratio of each label across the 4-step AMD severity classes. Labels include (a) drusen area, (b) maximum drusen size, (c) reticular drusen presence, (d) soft drusen type, (e) hyperpigmentation area, (f) depigmentation area, (g) total geographic atrophy (GA) area, and (h) central GA area. Category definitions for each fundus feature are shown in Figure 3.11.

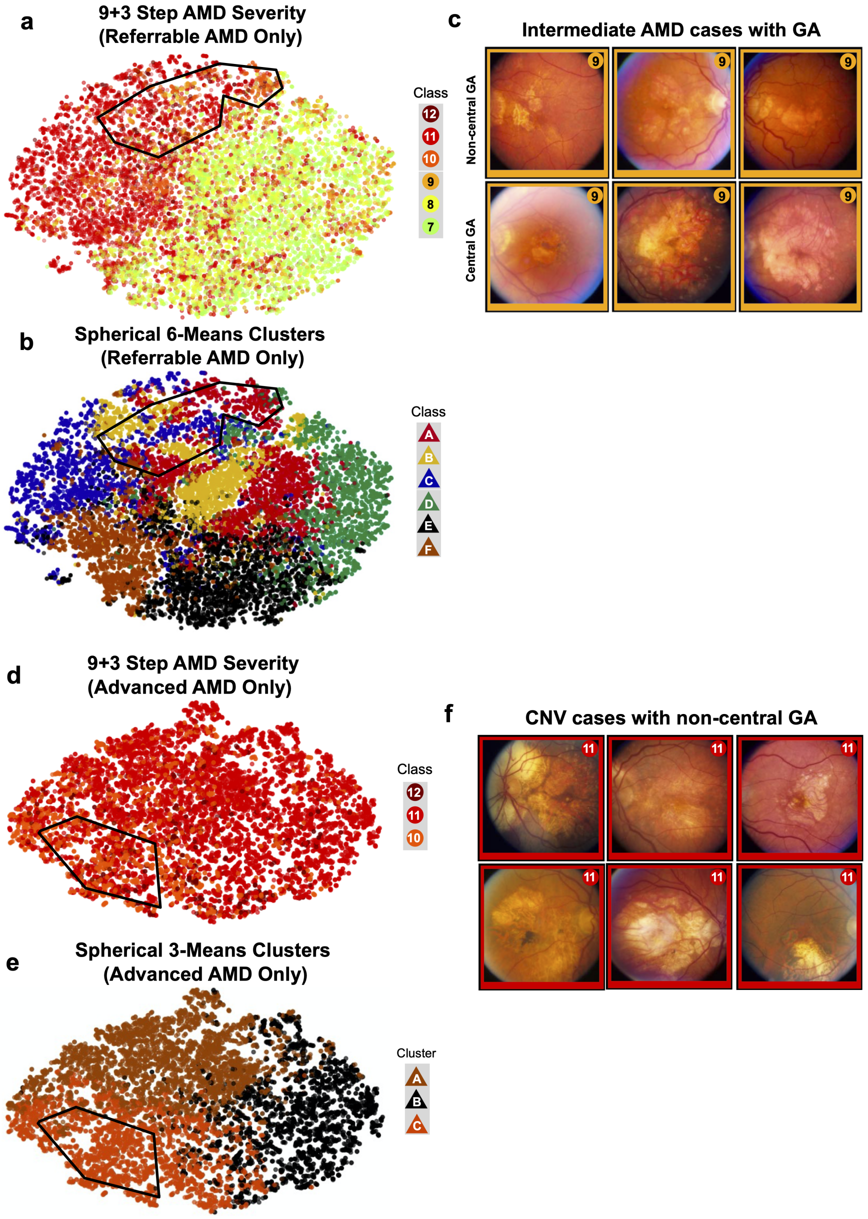


Figure 3.9: Data-driven discovery of central and non-central geographic atrophy. t-Distributed Stochastic Neighbor Embedding (t-SNE) visualizations of NPID feature vectors colored by (a) 9+3-step AMD severity labels and (b) spherical K-means cluster labels with K=6, based on hierarchical learning using only fundus images with referable AMD (intermediate or advanced AMD). A selection (outlined area) of intermediate AMD cases (classes 7-9) adjacent to advanced AMD cases (classes 10-12) from clusters A-C show (c) fundus images with non-central GA (top row) and central GA (bottom row). t-SNE visualizations of NPID feature vectors colored by (d) with 9+3-step AMD severity labels and (e) spherical K-means cluster labels with K=3, based on hierarchical learning using only fundus images with advanced AMD (classes 10-12). A selection (outlined area) of CNV cases (class 11) adjacent to images with central GA with or without CNV (classes 10 and 12) from cluster C show (f) non-central GA.

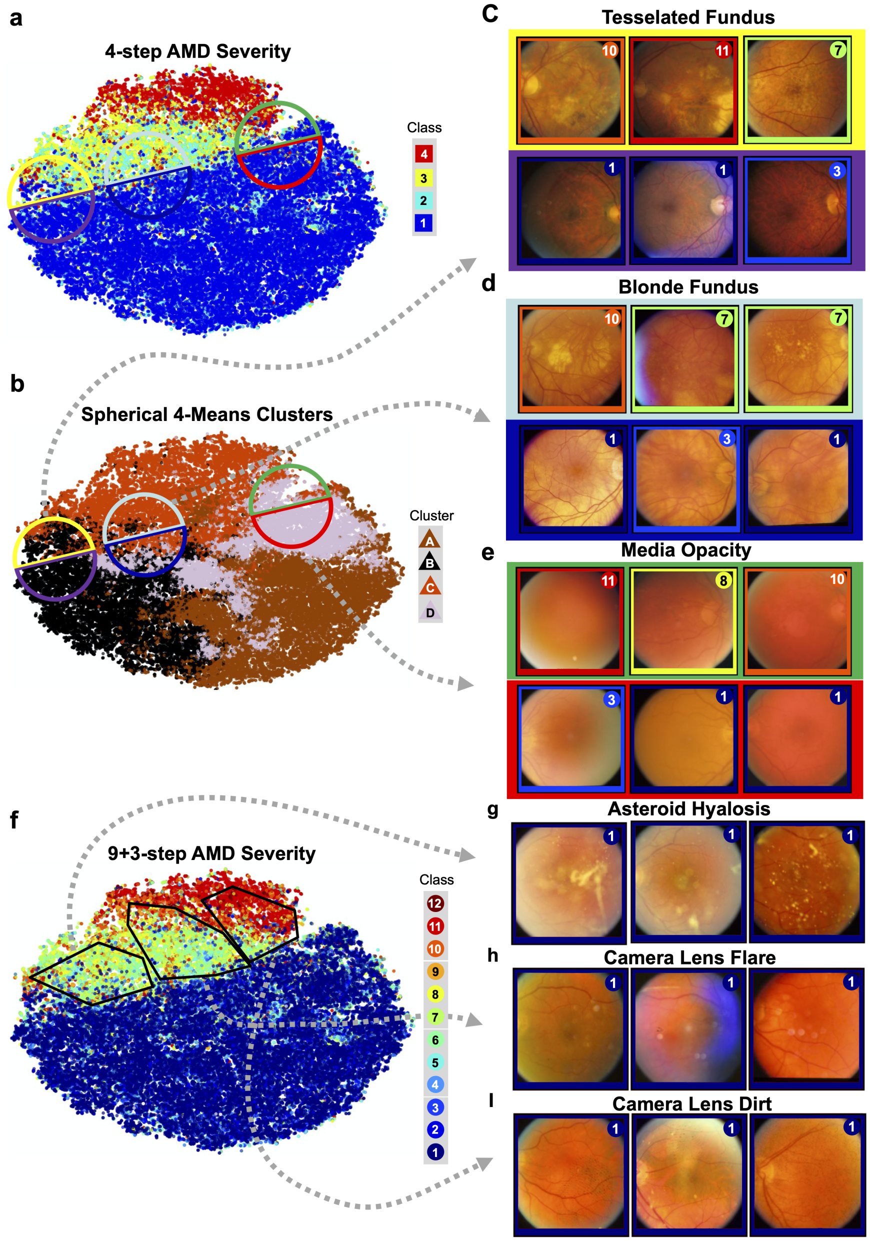
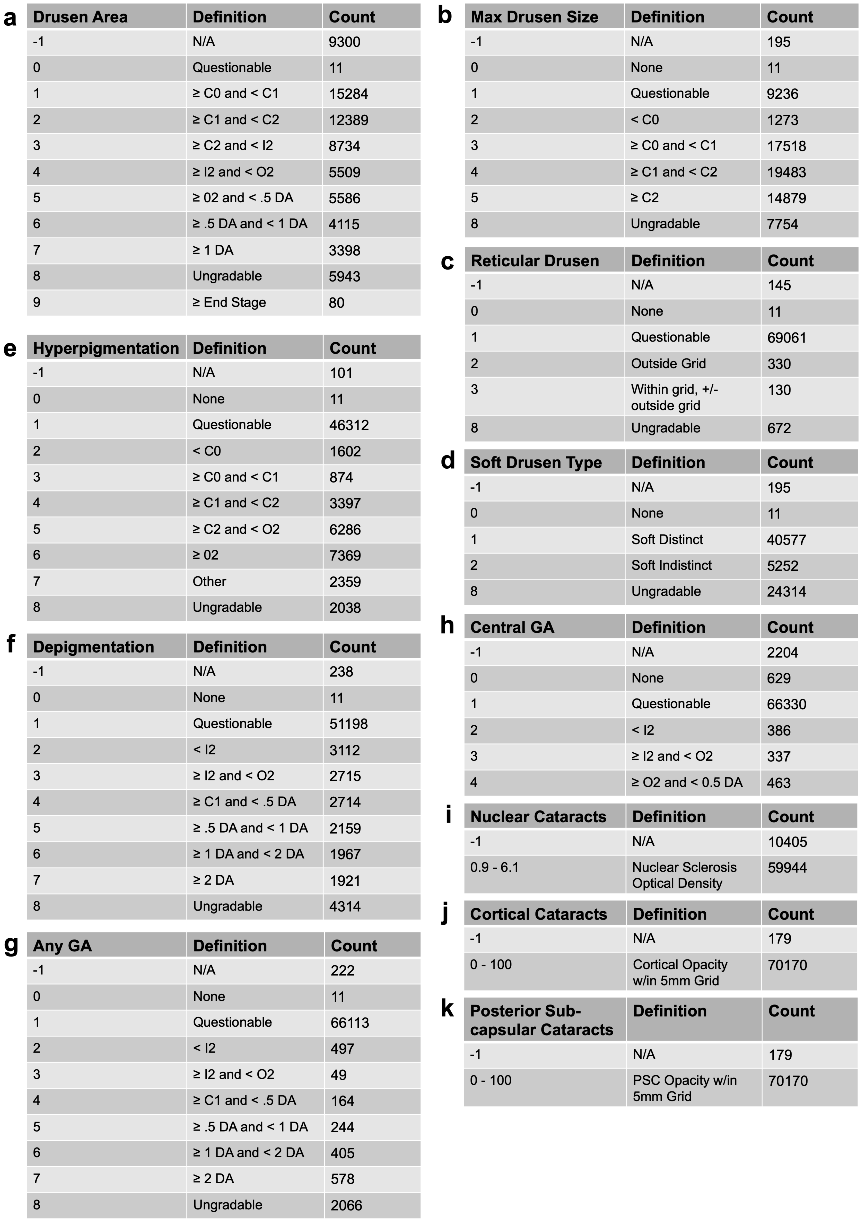


Figure 3.10: Data-driven discovery of ophthalmic features. t-Distributed Stochastic Neighbor Embedding (t-SNE) visualizations of NPID feature vectors colorerd by (a) 4-step AMD severity labels and (b) spherical k-means (K=4) cluster labels. Fundus images that straddle no AMD vs. early, intermediate, or advanced AMD within K-means cluster B (yellow-purple circle), cluster C (teal-blue circle), and cluster D (green-red circle), corresponded to fundus images with (c) tessellated fundus, (d) blonde fundus, and (e) media opacity. (f) t-SNE visualization of 9+3-step AMD severity labels with a selection (outlined areas) of fundus images with no AMD (class 1) located within clusters of early, intermediate, or late AMD classes corresponded to fundus images with (g) asteroid hyalosis, (h) camera lens flare, and (i) camera lens dirt.



a

Drusen Area	Definition	Count
-1	N/A	9300
0	Questionable	11
1	≥ C0 and < C1	15284
2	≥ C1 and < C2	12389
3	≥ C2 and < I2	8734
4	≥ I2 and < O2	5509
5	≥ 02 and < .5 DA	5586
6	≥ .5 DA and < 1 DA	4115
7	≥ 1 DA	3398
8	Ungradable	5943
9	≥ End Stage	80

e

Hyperpigmentation	Definition	Count
-1	N/A	101
0	None	11
1	Questionable	46312
2	< C0	1602
3	≥ C0 and < C1	874
4	≥ C1 and < C2	3397
5	≥ C2 and < O2	6286
6	≥ 02	7369
7	Other	2359
8	Ungradable	2038

f

Depigmentation	Definition	Count
-1	N/A	238
0	None	11
1	Questionable	51198
2	< I2	3112
3	≥ I2 and < O2	2715
4	≥ C1 and < .5 DA	2714
5	≥ .5 DA and < 1 DA	2159
6	≥ 1 DA and < 2 DA	1967
7	≥ 2 DA	1921
8	Ungradable	4314

g

Any GA	Definition	Count
-1	N/A	222
0	None	11
1	Questionable	66113
2	< I2	497
3	≥ I2 and < O2	49
4	≥ C1 and < .5 DA	164
5	≥ .5 DA and < 1 DA	244
6	≥ 1 DA and < 2 DA	405
7	≥ 2 DA	578
8	Ungradable	2066

b

Max Drusen Size	Definition	Count
-1	N/A	195
0	None	11
1	Questionable	9236
2	< C0	1273
3	≥ C0 and < C1	17518
4	≥ C1 and < C2	19483
5	≥ C2	14879
8	Ungradable	7754

c

Reticular Drusen	Definition	Count
-1	N/A	145
0	None	11
1	Questionable	69061
2	Outside Grid	330
3	Within grid, +/- outside grid	130
8	Ungradable	672

d

Soft Drusen Type	Definition	Count
-1	N/A	195
0	None	11
1	Soft Distinct	40577
2	Soft Indistinct	5252
8	Ungradable	24314

h

Central GA	Definition	Count
-1	N/A	2204
0	None	629
1	Questionable	66330
2	< I2	386
3	≥ I2 and < O2	337
4	≥ O2 and < 0.5 DA	463

i

Nuclear Cataracts	Definition	Count
-1	N/A	10405
0.9 - 6.1	Nuclear Sclerosis Optical Density	59944

j

Cortical Cataracts	Definition	Count
-1	N/A	179
0 - 100	Cortical Opacity w/in 5mm Grid	70170

k

Posterior Sub-capsular Cataracts	Definition	Count
-1	N/A	179
0 - 100	PSC Opacity w/in 5mm Grid	70170

Figure 3.11: Class definitions for AREDS Reading Center labels. Descriptive tables detailing label definitions for (a) drusen area, (b) max drusen size, (c) reticular drusen presence, (d) soft drusen type, (e) hyperpigmentation area, (f) depigmentation area, (g) total geographic atrophy (GA) area, (h) central GA area, (i) nuclear cataracts severity, (j) cortical cataracts severity, (k) posterior sub-capsular cataracts severity. In the table definitions, C, I, and O correspond to groups of open circles, where C=Central, I=Inner, and O=Outer. Their numbers correspond to the Disc Diameter (DD) in relation to the average Disc Area (DA), C0=0.042 DD, C1=0.083 DD, C2=0.167 DD, I1=0.120 DD, I2=0.241 DD, O1=0.219 DD, O2=0.439 DD.

Chapter 4

Self-Supervised Representations for Zebra Finch Audio Data

4.1 Introduction

We have shown evidence that NPID can functionally train a CNN to perform a task aside from object classification on images with natural statistics, where the actual task requires medical expertise to perform classification. We also showed that investigating the learned features reveals insights into the underlying medical data: (a) data-driven learning without expert guidance reduces bias in disease classification, and (b) cluster-analysis tools can aid clinicians discover expert-level groupings across a range of medical tasks on the same data. We now want to evaluate NPID's use in another domain that requires biological expertise across a non-natural statistical distribution of data: zebra finch audio vocalizations.

Audio is a different modality than vision, so it is not obvious whether we can use NPID to train a network to discriminate task-relevant features, or if we need expert-level guidance. Furthermore, audio samples are 1D waveform signals, but we want to extract more information by converting the inputs into 2D images. While structured representation learning is necessary for learning on both modalities, one difference between vision and audio is the level of contrast between signal and noise. How will NPID organize features and how can we probe representations without other feature labels?

Supervised Audio Processing

Supervised learning has been applied to biological audio data in the past, with reasonable classification performance using older machine learning techniques but with clear limitations. The first is that most research studies perform supervised learning for species classification, like for for birds, bats, or dolphins sounds in the wild [73, 28, 115, 41]. This implies that the level of discriminability learned across audio samples cannot separate call types within a species. The second limitation is that even if a study is performing intra-species classification of audio samples, their best results come from extracting expertly-defined audio features and learning to discriminate in this predefined feature space [37]. This approach moves away from biologically-inspired learning, which can both data-driven and expert-driven. Furthermore, supervision constrains the learning space to a fixed set of boundaries for call representations.

CNNs for Audio Processing

Deep learning is being applied in this domain more recently. Specifically, sound event localization and detection with neural networks in general is gaining interest in the research community [96]. Sound as input modality can help in robotic navigation tasks either to navigate towards sounding objects [26, 25, 24], provide layout information [29, 100, 46], or to tracking and segment sound events, even behind audio occlusions [125, 45]. Impact sound can also be leveraged for object and material classification [23, 143]. In videos, the audio-visual signal allows to separate individual voices or musical instruments [47, 121], and improves action recognition as computationally favorable modality providing additional features [134, 68].

Data-driven Audio Learning

Self-supervised and unsupervised learning organizes stimuli based on features that are not predetermined by human labeling. Unsupervised and semi-supervised neural networks have been developed using several methods, including instance-based learning, exemplar learning, deep clustering, and contrastive learning [36, 9, 142, 92]. As a contrastive learning approach, Non-Parametric Instance Discrimination (NPID) was previously designed for complex visual tasks [133]. Networks trained with NPID first learn to identify each stimulus as being uniquely encoded compared to every other stimulus, so each image's feature vector gets pushed toward its corresponding feature vector in the previous epoch, and away from many other negative samples. As long as the feature encodings are not too high-dimensional, this contrastive push and pull creates distinct neighborhoods of training feature vectors that are task-relevant, and can subsequently be used for supervised evaluation. After that initial training, NPID predicts a stimulus class label by determining the most common label among its nearest neighbors within a multi-dimensional hypersphere of encoded feature vectors drawn from training stimuli. In addition, data augmentation can boost NPID training performance across training epochs by enforcing a level of transformation invariance (e.g. separately cropped views of the image represent positional invariance). In previous work, this technique significantly outperforms other unsupervised networks for ImageNet, Places, and PASCAL Visual Object Classes classification tasks [133].

In general, NPID features are primarily learned through contrastive repelling of negative pairs of instances. Better contrastive learning occurs with more negative samples [122, 60], but that is limited by computational memory. Data augmentation effective creates more negative samples for contrastive learning. Further, data augmentation helps create transformation-invariant encodings, which are harder to repel using only low-level image differences. This encourages common repulsion of samples from the same class away from samples from other classes. However, more solutions than data augmentation are possible for better contrastive learning. An updated version of NPID attempts to modulate the distance between the negative pairs based on presumed cross-level hierarchy of instances and groups [130].

CNNs for Biological Audio Data Like imagery data (e.g., camera trap images), the ecological community has also actively applied deep learning to bioacoustics for all sorts of applications. For example, using deep learning to detect and identify animal species from audio clips (e.g., bird species and marine animals) [67, 59, 2, 147, 146, 12, 113]. In addition, using deep learning methods to classify and understand animal behaviors and functional calls [101, 145, 85]. However, most previous studies focus on supervised machine learning, which requires a large number of ground-truthed annotations for the training process. Thus, to explore methods that do not rely on annotations, some studies also introduce unsupervised learning methods (such as Gaussian Mixture Models and Auto Encoders) to cluster bioacoustics data without annotations [85, 89, 31]. However, previous unsupervised bioacoustics studies are either based on relatively simple tasks (e.g., clustering between two independent categories, fish and whale) [89] or have limited data sizes and performance [31].

Audio Representations in the Brain

Vocal communication plays a central role in coordinating the behavior of social animals. Many vertebrate species, such as the spotted hyena [73], the African elephant [115], the vervet monkey [28] and a myriad of bird species [41, 32, 37], use a large repertoire of call-types in clearly distinct behavioral situations to signal danger, aggression, distress and hunger or to encourage cooperation, pair-bonding, and mating intentions. This repertoire of call-types has been called the language" or animals [80]. Although comparisons between human language and animal vocal communication quickly become controversial as one addresses higher linguistic functions [109], it is clear that some of the computational issues faced by computers for automatic speech recognition and by the brain of humans and other animals are similar: the identity of the words in the speech stream or of the call-types in an animals' call exchanges needs to be determined. This sound categorization task is made possible because of each phonemic unit or each call-type is characterized by a set of identifying acoustical features. However, it is also a difficult task because these identifying acoustical features can be "high-level" (e.g. formant transitions for distinguishing among stop consonants) and because of the acoustic variability that is found across vocalizers [38], across renditions from the same vocalizer (e.g. in different states emotional states)[94], and because information bearing structure in the sound can be degraded by propagation [86] or contamination by other sound sources [84]. Thus, in order to understand the computational steps that allow computer algorithms or brains to recognize words in the speech stream or vocal-types in vocal exchanges in animals it is essential to, first, describe the nature of the identifying acoustic features and, second, to understand how this acoustic code can be discovered by a computer or a brain amidst the variability found in natural soundscapes.

Researchers in human phonetics or animal bioacoustics have made significant progress in the first task: describing potential acoustic codes that could be used to discriminate among the group of sounds making these meaningful categories [16, 126, 111, 55]. At the same time neurophysiologists have made some progress in investigating the neural representation of these call categories and in doing so determining whether the acoustical features used by the brain are similar to those found in bio acoustical analyses.

The sound features extracted by high-level single auditory neurons have been described by their single or multiple spectro-temporal receptive fields [120, 6, 70] and the features extracted by these receptive fields can be related to specific acoustical structure found in natural sounds [63, 132, 103, 21].

Much less is known about the second task in the biological systems: the discovery of the set of relevant and maybe even optimal high-level features for performing the categorization. In humans, the developmental stages of speech perception have been well described [74] but the underlying brain changes that occur in speech areas during learning and development remain unknown. In animals, the recognition of identifying features can be innate, in which case this discovery phase occurs on an evolutionary time scale [48], or can be learned by experience as in humans [110]. Ultimately, however, we are do not know how the evolved or learned neural representation for high-levels auditory features that could be used to categorize vocalizations into their meaningful units emerge in brain systems. It is in this domain that one can obtain significant insights by leveraging the power of machine learning approaches to automatic sound recognition. More precisely, we propose to analyze the features extracted by CNNs trained at a call-type discrimination task and compare them to those that have be obtained in neurophysiological recordings in response to the same stimulus set.

We will test the hypothesis that efficient representations of high-level acoustic features in CNN and auditory systems can arise by statistical learning: the repeated exposure to the underlying statistical structure of noisy stimuli is sufficient to generate high-level representations that capture the underlying structure (e.g. categories) of the stimulus [42]. More specifically, we will test the performance in the call-categorization task by an self-supervised CNN and compare the acoustic features extracted by such CNN to those obtained in high-level auditory areas. We will perform this analysis using the vocal communication system of a social songbird, the zebra finch which we have studied using bioacoustical, behavioral and neurophysiological analyses. The self-supervised CNN used will be based on a deep convolutional network architecture trained by a form of instance-based learning procedure called Non-Parametric Instance Discrimination (NPID) [133].

Audio Representations in Zebra Finches

In this study, we used a CNN backbone trained via NPID on the large data base of call-types of the zebra finch repertoire described in [37]. We first tested whether the representation extracted in the output layer of the unsupervised CNN could be used for classifying novel-call types based on the nearest neighbor voting scheme and whether the classification performance of this CNN would match or exceed the one that was obtained with supervised classifiers or by decoding the neural activity. Second, we assessed the nature of the extracted sound feature representation, by describing the responses of the output layers in terms of their MTF. We compared these artificial MTFs to the neuronal MRFs that we obtained in our neurophysiological data to determine whether neural responses in high-level auditory areas could have been learned simply by statistical learning implemented as an instance discrimination.

Receptive Fields are a descriptor of what input pattern a given learned filter is tuned toward. These were originally created for neuroscientific experiments to evaluate how successive layers of the cortex process different hierarchies of features. A neuron's behavior (through recordings like fMRI, EEG, cell spike activity, etc.) can be correlated with different measurements of input features. Assuming sampling happens over a well-distributed set of inputs, we can estimate which input features drive changes in neuronal output activity. Similar receptive fields have also been generated for neural network analysis, both for well-defined input features across many domains of features [8], as well as for class-correlated input features [148].

Ultimately, we want to (1) evaluate if and how a data-driven CNN can learn salient representations for this audio task, with biologically-plausible input data, and (2) quantify receptive fields for output neurons and evaluate if these learned representations compare to those from neural response models for the same task.

Zebra Finch Audio Dataset

Zebra finches are highly intelligent, social birds that depend on functional calls to one another. Even as chicks, they interact and mature with groups of up to 100 zebra finches, so they are able to distinguish and communicate with individuals across time. Domesticated zebra finches also reproduce almost every single call a wild zebra finch would (except for a group, migratory call). This means that zebra finches are a valuable research animal for understanding how lower-level vocalizations can relay and be processed into higher-level concepts.

The audio dataset across domesticated zebra finches was collected by Elie and Theunissen [37]. There are 3433 stereo audio waveforms with stereo channels across the 12 call types, which vary in temporal amplitude envelope (i.e., the limits of the waveform shape over time), intensity, and frequency, all visible through the waveforms in Fig 4.1. For instance, some classes (e.g., Te, Th, Tu) have samples much shorter than others (e.g., DC, Ag, So), so we would expect higher frequencies to be more relatively salient for representing the first group. Alternately, even across classes with long samples (e.g., Be, Ag, Di), we see differences in intensity and frequency distribution.

Fundamentally, a challenge of this dataset is that it is imbalanced from all perspectives. Even seen in Figs 4.1 and 4.1, the distribution of waveform counts and call length per functional call all long-tailed distributed across call type, age, and individual. Can NPID functionally separate these different distributions of inputs?

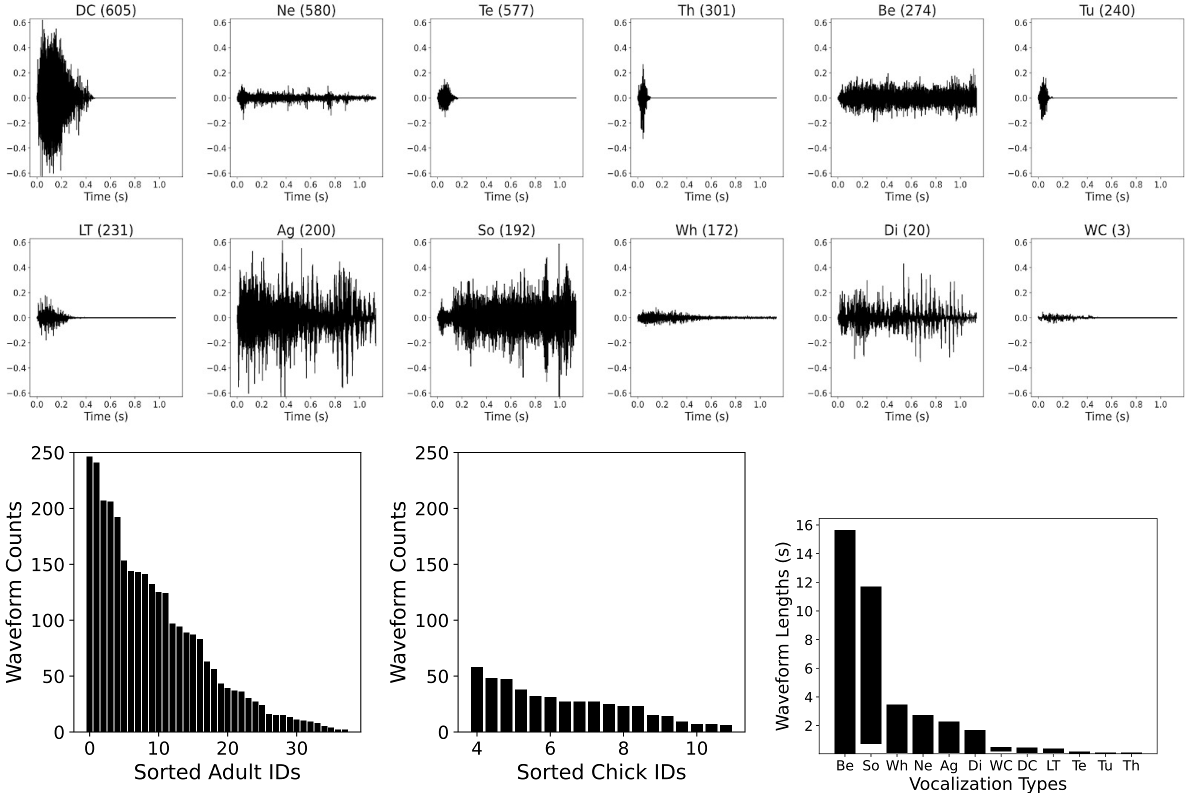


Figure 4.1: Data Plots for Zebra Finch Vocalizations. (Top) Representative samples from each vocalization class, sorted by sample count per class (counts in parentheses). (Bottom left) Histograms showing sample counts per individual for adult and chick zebra finches. (Bottom right) Sorted bar plot showing min and max waveform lengths per vocalization class

Spectrograms

Even without assessing empirical performance, spectrograms are a theoretically salient input representation for audio classification, and have been commonly used [67, 59, 146] for deep learning experiments. Mathematically, spectrograms are short-time Fourier transforms (STFTs) that calculate spectral information within overlapping temporal windows; each ”column” in the spectrogram image represents the frequency distribution of one time window. As a whole, spectrograms visualize a distribution of frequencies across a given waveform, and 2D spectrogram activity visually matches 1D waveform activity over time Figure 4.2.

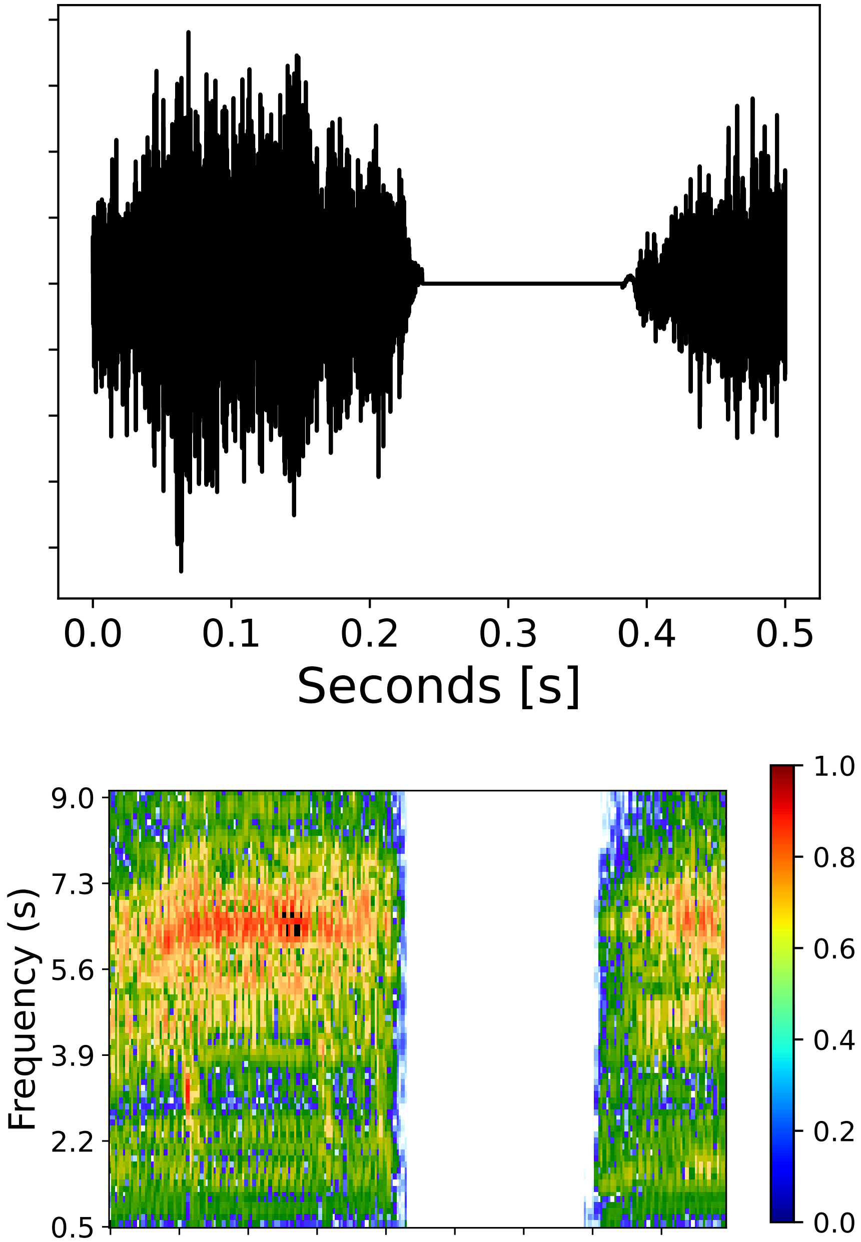


Figure 4.2: Sample Waveform and Spectrogram. Representative Sample Waveform and Corresponding Spectrogram for a 0.5 second audio clip from the 'Be' ('Begging') call type

When interpreting a spectrogram, local energy peaks are referred to as formants, and they indicate that band of frequencies form a dominant part of the incoming sound. These formants can be described by the audio features they impact. Two common features for speech recognition are pitch and timbre. Pitch is easier to understand, as most people can intuitively match a given tone to an frequency range (i.e. low, mid, or high), though the same tonality can be equated by different harmonics of the same fundamental frequency. Timbre is more caused by the larynx or vocal chord. Consequently, two people can speak at the same pitch with different timbres (e.g. consider Barack Obama and Arnold Schwarzenegger trying to match the same pitch – they will still sound different). The individual differences in their throats and mouths create different vowel sounds, which can be as salient to specific vocalized call types as it can be different individuals.

When assessing zebra finches, we see more vocal similarities than differences across individual birds, so we expect timbre to be salient to functional vocalizations.

Modulation Power Spectrum

The Modulation Power Spectrum (MPS) of a given audio signal is another representation that shows temporal dynamics of pitch and timbre. However, unlike a spectrogram, it is not sensitive to start time. It is computed as a spatially-weighted average of overlapping windows across a spectrogram, specifically with a 2D gaussian weighting. For simplicity, the MPS can be equated to a second Fourier transform applied on the STFT of a given waveform, even though that is not entirely accurate.

This new audio representation shows the dominant temporal frequencies in a given signal, independent of when they are, while also representing how those frequencies shift over time. When a given energy band is more asymmetric to the left of the y-axis, we describe this as an an ”upsweep”, and the opposite as a ”downsweep”.

We can assess the periodicity of a signal's pitch along the y-axis of the MPS. Note, that the frequency representation is inverted. 2cyc/kHz corresponds to 500Kz, and energy bands above it represent lower frequency, while energy bands below it represent higher frequency. The timbre of a signal is described by the formant information just above the x-axis, while the x-axis describes the rhythm of the signal 4.3.

Just as a spectrogram represents low-level information akin to low-level cochlear processing, the MPS can be viewed as representing mid-level audio features. One example feature is tonality, which can be viewed analogous to mid-level visual features, like convexity. This may not be directly tied to spectrogram features, but they derive from them.

It is relevant to establish the baseline features that can be represented in this feature space. The two important baselines are the average MPS per sample, and the average MPS per class. The average MPS signal per sample (seen in Fig 4.4 is subtracted from every sample's MPS, so we assess an input relative to the calculated average. This means we analyze a model's response sensitivity relative to a perceived common experience, which is more relevant to feature extraction (e.g. the redder the apple compared to raw green, the riper it is). The same feature assumptions are for true class-average MPS plots, so sample averages are removed from class averages (Fig 4.5) to assess the function of neuronal receptive fields in that new class-average space.

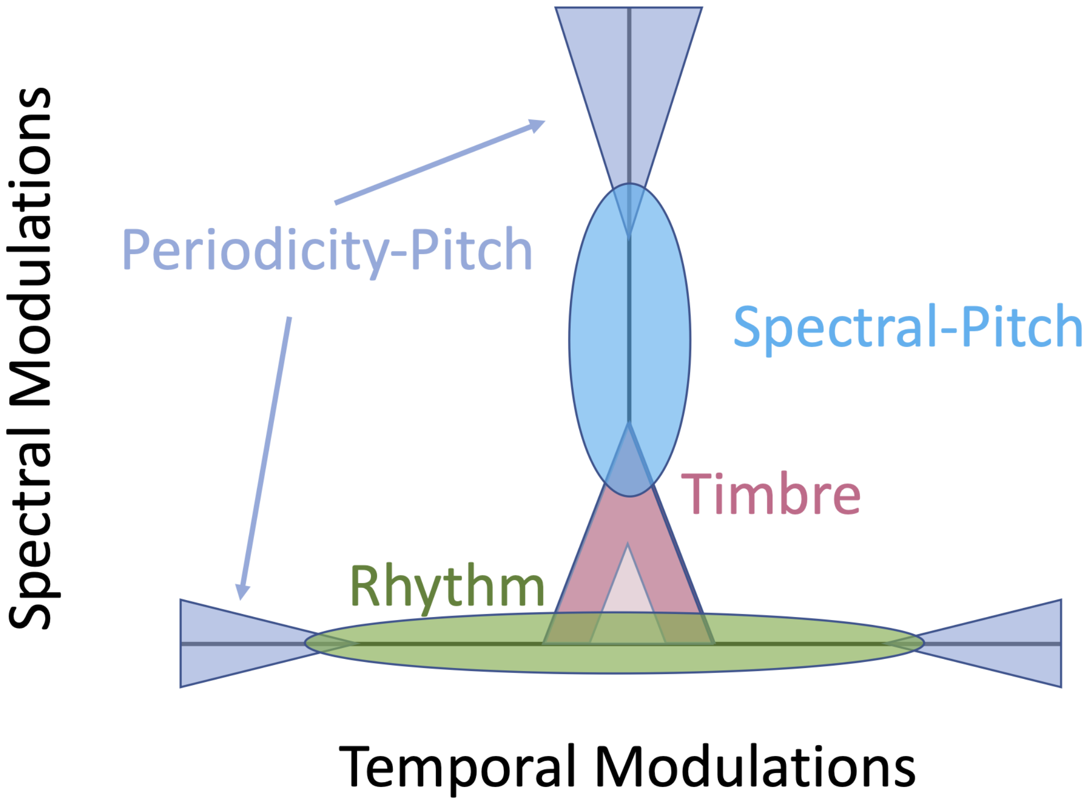


Figure 4.3: Diagram of Modulation Power Spectrum Features. The interpretation of audio features from MPS representations is different than for spectrograms. Here, the x-axis represents temporal modulations along the spectrogram, while the y-axis represents spectral modulations. The main areas of focus will be the pitch along the y-axis, and the timbre formants above the origin.

4.2 Methods

Preprocessing

At each epoch of training or testing, input samples are again preprocessed, but our model's validation accuracy derives from a low amount of preprocessing. Contrary to standard deep learning procedure, common applied transformations did not boost performance, even ones that were designed for audio learning (not shown). For a published supervised deep learning model, they found that applying transformations directly on the spectrogram reduced error rate for word classification [90]. They applied time masking (masking spectrogram rows with fixed values), frequency masking (masking spectrogram columns with fixed values), and time warping to learn representations that are invariant to occlusions and speed changes in audio

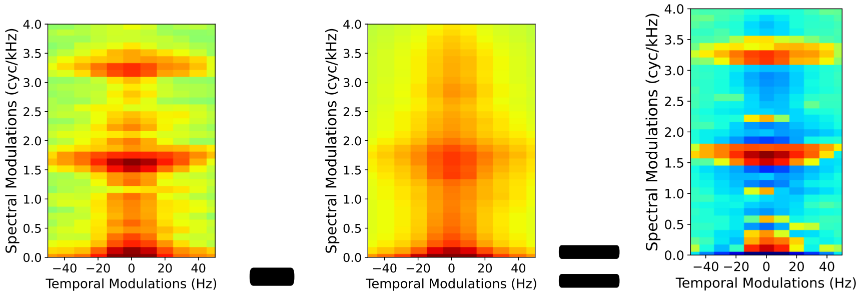


Figure 4.4: Mean-Subtracted MPS. Representative plot showing the average MPS removed from a sample's MPS so the final difference can be analyzed without bias. It is important to assess differences relative to the average, as neural preferences are relative not absolute.

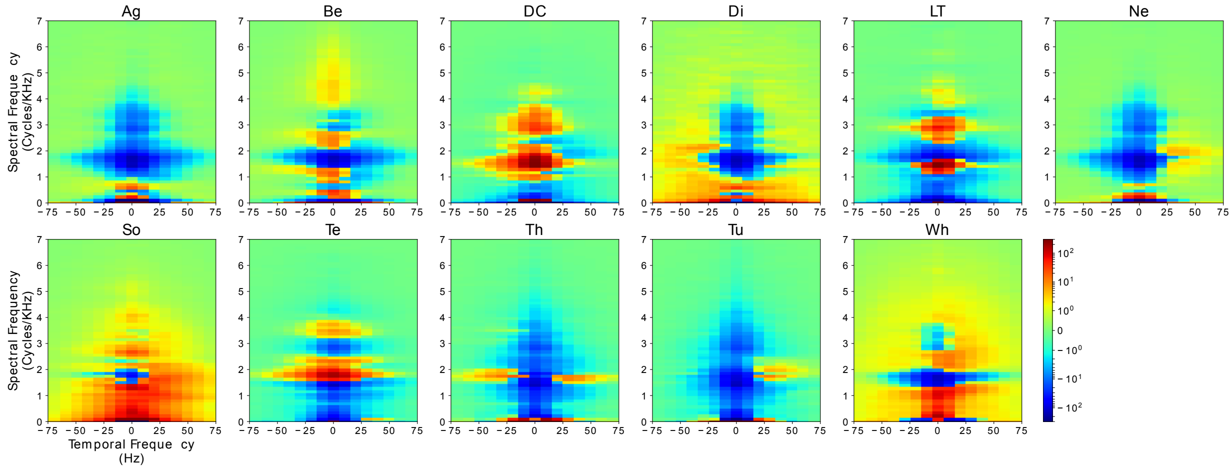


Figure 4.5: Average MPS Per Vocalization Class. Average MPS plotted per vocalization class, with with the same colorbar limits. Here, the average MPS per sample is removed for each class plot, as it is important to assess differences relative to the average, since neural preferences are relative not absolute

data. We attempt to explain this rejection of transformation viability in the Discussion.

Most of the pre-processing is fixed with each iteration. First, a 1D input waveform is cropped to .5s and converted to a 2D input spectrogram. Even though STFTs compute both the magnitude and complex phase in the Fourier domain, we ignore the angle information. We could not improve performance with different ways to include phase in the input (not shown). Second, the spectrogram magnitude is log-scaled to better visualize energy contrast. Third, bandpass filtering was performed directly on the spectrograms. This was chosen as filtering raw waveforms would be too complicated by creating aliasing effects that need to be extrapolated away). Finally, the 1-channel magnitude input was mapped to a 3-channel color before training.

As an additional note, the spectrogram shown in Figure 4.2 visualizes the magnitude with an RGB color map. While this representation may not exactly match the input for biological auditory processing, it is holistically comparable. The ear separates an audio signal through successively lower-pass filters, which ends up breaking down a sound signal into band-passed signals across multiple bins of frequency filtering, and this frequency binning is analyzed over time [79]. As such, this biological input pipeline compares similarly to that of a spectrogram input.

The variable part of the preprocessing pipeline should be thought of as data augmentation. As implied by recent experiments at Google, data augmentation can be necessary for learning to discriminate audio samples (especially individual semantic units), since audio data is generally limited [90]. We also know that data augmentation boosts NPID performance [133] in general, so we expect data augmentation to be useful in data-driven audio processing. Even though traditional data augmentation techniques did not work for our pipeline, we still need to tackle the problem of minimal data. We augmented samples by doing randomized time crops for our first preprocessing step.

Randomized time crops allows us to capture a signal at different start times, which works for samples that are both shorter and longer than the fixed window for cropping. If a sample is shorter than the time window, we include the whole signal and randomize ts start time. If a sample is longer than the time window, we just take a random, fixed subset of the waveform. This enforces a level of temporal invariance to learned features for each individual sample. This way, our model does not expect the ”start” to a call to always be in the left part of the spectrogram, or to always expect to experience the whole call to identify it.

In general, NPID is still susceptible to data imbalance, so the fixed-window time cropping helped our model assess a more uniform distribution of temporal dynamics across samples, without ignoring any data.

NPID Training and Evaluation

NPID training and testing is accomplished with the same general pipeline as described in Chapter 3, as seen in Fig 3.5. For sake of brevity, this pipeline is not repeated in detail. Input samples are preproccesed and fed into a ResNet-50 backbone for training, and output encoding vectors are normalized with L2 normalization (i.e., mapped to unit vector representations). Each training sample is encoded into unit feature vectors and stored in a memory bank for use across epochs. Instance discrimination is learned through contrastive learning of differently preprocessed inputs. Positive pairs correspond to the same input sample across epochs, and negative pairs correspond to different input samples. This contrastive push and pull is weighted based on distance, where distances equal angles for unit vectors. This weight is thus an softmax of cosine similarity for pairs compared to all pairs. For more detail, please reference the original NPID training paper [133].

The memory bank of training vectors is also used in NPID evaluation. Classification with NPID-learned features depends on a voting scheme between nearest encoded training samples and a given query called weighted kNN voting (w-kNN). Query inputs are encoded through the NPID pipeline, and its distance (i.e., cosine similarity) is computed across k-nearest neighbors of training vectors. Each training retrieval is assigned its expert label for the task, and retrieved labels are tallied for the prediction of the query label. Just as the loss function for training, the voting scheme for evaluation is exponentially weighted based on distance, and a temperature hyperparameter determines just how exponentially weighted the voting is. Just as in Chapter 3, the hyperparameter k for w-kNN voting is empirically tuned across range of values between 5 and 50.

Analogous to t-SNE visualizations from Project 2, we generate Uniform Manifold Approximation and Projection (UMAP) visualizations for our training feature space. UMAP approximations reduce dimensionality while preserving local topology just like t-SNE approxmiations, but also maintains more global structure, unlike t-SNE.

To compare with previously published data, we estimated the posterior probability alongside class accuracy from w-kNN voting. The posterior probability estimates the likelihoods of classifying each class given that the sample is from a given class, and relates to the confusion matrix. We computed posterior probabilities using the same exponentally-weighted cosine similarity of positive pairs to nearest neighbors from each class. To balance the prior distributions, we normalized over an equal distribution of neighors from each class.

Lastly, 10-fold training and validation allowed us to evaluate generalizability with the limited data available.

[**Receptive Field Generation and Evaluation**] We generate novel receptive fields in the MPS feature space using a common receptive field pipeline. For a given output neuron from our neural network, we take a weighted average of all input MPS, where the weights are the output activation for that neuron. We refer to this receptive field as an MTF. We also generated biological receptive fields from neuronal spike data from zebra finch recordings, and refer to these as MRFs. Artificial MTFs were compared with biological MRFs, both relative to the average MPS of all inputs. Because MTFs and MRFs are in same MPS space defined by the same limits, we can compare the two receptive fields, for which we also use cosine similarity.

To assess MTF representational salience relative to neuronal preferences, we performed principal component analysis (PCA) on all the artificial MTFs. The principal components (PCs) were generated in the MPS space. Using the top 3 principal components as bases, we projected MTFs and sample MPSes into this orthogonally-defined space, and visually compared distributions.

4.3 Results and Discussion

Classification Performance and Retrieval Explainability

Our hyperparameter selection for w-kNN voting was empirically decided based on results from crossfold training. For each run, classification accuracy was computed across all listed k (5-50), and the k with the highest accuracy was chosen. Though not shown, this k=8.

Fig 4.6 shows class confusions for the run with 66% average class accuracy on validation data, with the sample average being 68%. Notably, most classes are well-discriminable, with Tu having the lowest class score of 35% and So having the highest class score of 97%. What is additionally interesting is that the Di call was vastly underepresented compared to other calls (Fig 4.1), but still outperformed Tu.

Furthermore, misclassification can be explained by the confusion matrix. For example, poor Tu classification performance can be explained by class-confusion with Th, which is still biologically plausible. Functionally, Tu and Th are variants of the same alarm call type, so they can be viewed as sub-classes of the same super-class. However, we know from Chapter 3 that NPID is not suitable for fine-grain classification of sub-classes, so we have likely achieved near optimal discriminablity without modifying our data-driven process. In addition, LT misclassifications with DC can also be biologically explained. LT and DC are both variants of the same distance call, but across chicks and adults. Plausible explanations are either (a) some learned representations for LT are invariant to pitch differences between LT and DC, or (b) some LT data is more fundamentally similar to DC data than other LT data.

Many class confusions can be explained by the UMAP visualizations of the encoded training vectors. Fig 4.6 shows how all the training samples are distributed, color-coded by functional class. Most notable are our two confusion pairs. Th and Tu overlap almost entirely, so mapping them to the same class is reasonable. However, the LT and DC distributions have only partial overlap. This partial overlap explains the partial confusion between LT and DC.

Additional explanations for LT and DC confusions can be extrapolated from data plots. Fig 4.1 shows that DC is vastly overrepresented compared to LT, so better DC classification and more LT misclassification can be attributed to w-kNN bias derived from these distributional differences. Similarly, Fig 4.5 shows the average difference between DC and LT signals in the MPS space. LT distribution is very similar to DC's distribution, but with relatively less energy at around 2cyc/kHz (500Hz signal frequency). This combined with misclassification results suggest that some LT samples are closer to the average DC distribution than the average LT distribution.

From Fig 4.7, we can see our best run's posterior probability matrix compared to the published baseline [37]. The average probability was 56%, which approaches performance from the published baseline classifiers from neural responses on the same data. This difference in average probability may be explained by different prior estimates per class, and different priors may need to be evaluated.

Similar to Chapter 3, we present example retrievals for correct and incorrect classifications. The advantage to using NPID over deep learning algorithms without memory banks is we can visualize the exact images that drive a given prediction. Even though the underlying features are learned from back-propagation just like every other network, neighborhoods of features ultimately determine task performance, and those neighborhoods can be visualized.

Correct prediction retrievals show similar structures between training references and query images. Fig 3.7 shows how Be, So, and Te samples can be correctly predicted by spectrogram feature encoding, and these encoded features are visually separable. For example, Te class samples can be described by single lower frequency energy bursts, whereas Be and So can be described by 3 and 7 sound events, respectively, at higher frequencies.

Failed prediction retrievals also provide insights into the vocalization prediction task. Fig 3.7 shows Th appears similar to Tu samples with less higher frequency energy. Similarly, Be samples with more background noise appear indistinguishable from Ne samples. These classes do tend to have longer waveforms (Fig 4.1), so longer time crops of waveforms may improve separability of these classes, potentially at the expense of classes with shorter samples. A future direction may be toward a model that generalizes across input temporal lengths.

Lastly, with respect to failed classification retrievals, we see evidence for LT confusion with DC that aligns with class MPS and UMAP visualization differences. Based on Fig 4.5, we expect LT samples to have relatively less higher frequency energy, but some DC samples have similarly less higher frequency energy. Since DC samples overrepresent the training space, we expect these outlier DC samples to bias LT predictions.

These retrievals also provide evidence for why various transformations were not useful as data augmentation for NPID training. This example of Be and So sample confusion could also explain why time masking did not improve performance. Masking a Be sample could mask the sample appear to have more sound events like So, and masking an So sample could mask over sound events that would separate Be features from So features. Similarly, Ne and Be class similarities could explain why blurring or random noise addition was not useful, as they only increase similarities between the two. In addition, frequency masking could mask over frequency differences between LT and DC samples, leading to further misclassification. In general, masking and noise increasing transformations only increase sample similarity by attenuating differences.

Receptive Field Analysis of MPS Features

Fig 4.9 shows 20 of the 32 MTFs (i.e., artificial receptive fields) generated as output activation weighted averages of input MPS representations. Our assumption is that neural preferences are positively correlated with output activations, so we want to visualize what MPS features are correlated with the average input that drives a neuron's activation. We aim to evaluate if these MPS-defined receptive fields functionally describe neural preferences.

Early investigation of these MTFs seem to indicate response separability that useful for vocalization discriminability. We index rows and columns starting at 1 for reference. For

Row 1, Column 3 neuron, its MTF appears sensitive to dual harmonics at 1.5 cyc/kHz and 2 cyc/kHz, which is a plausible MPS of zebra finches sounds as they have separable left- and right-halves to their larynxes that allow them to create two pitches at once.

Further, the neuron at Row 2, Column 2 has an MTF that exibits harmonics at 1.5 cyc/kHz that could be useful for separation of Be or Wh call types from others. In addition, the neuron at Row 2, Column 5 appears to have the opposite sensitivity of features to the neuron at Row 3, Column 5. The former MTF is sensitive to higher pitch but lower timbre formants than the latter (note the inversion of spectral frequency at the y-axis compared to spectrograms). This could help separate low frequency and high frequency features for vocalizations with different vowel sounds.

We further compared these artificial MTFs with analogous biological MRFs generated from zebra finch neuronal response data. Of the 364 zebra finch neurons whose activity was recorded, the MRFs with the highest cosine similarity to artificial MTFs was discovered. Fig 4.10 shows 20 MRFs, each corresponding to its most similar MTF in the same grid locations as in Fig 4.9. We see staggering similarity in harmonics and general positive/negative selectivity of different energies, especially for the individual MTFs called out previously.

Based on visual analysis of MTFs and MRFs, these MTFs appear relevant for zebra finch vocalization classifcation. PCA should help us investigate if these MTFs actually encode a range of features that are salient for this task, and if we can interpret neuron behavior through MTFs. On face value, Fig 4.16 shows PCs of both domains of receptive fields being structurally similar.

Fig 4.11 shows the projection of each MTF in red onto the top 3 PCs (axes), as well as showing the projection of each sample MPS in blue onto the same PCs. The distribution of MTFs matches the overall shape of the distribution of MPS samples, so these MTFs relate to the input features encoded in this space. Further, Fig 4.12 shows separation of functional vocalization, color-code samples by their ground truth class. Not only is this distribution of encoded input features correlated with MTF response vectors along these orthogonal bases in the MPS space, the features are functionally separable in this space. This is further evidence that neuronal behavior can be described by MTFs.

In order to dive into the results, we also independently assess neuron MTFs. Individual analysis of each neuron reveals three classes of behavior along these functional bases in the MPS space. Figs 4.13, 4.14, and 4.15 show the vector projection of individual MTFs onto the PC bases. Each figure also shows MPS projections of average input samples across the 5, 10, 20, and 40 samples that either maximally (red) activate or minimally (blue) activate the corresponding neuron. Some neurons' behavior can be linearly described by these bases (e.g., Fig 4.13), as we can see the MTF projection vector lines up with the maximally and minimally activating sample projects. Other neuron behaviors can be quadratically described in this space (e.g., Fig 4.14), while others altogether require other non-linear descriptions (e.g., Fig 4.13). These results imply that the PC bases of MTFs is an appropriate starting point of quantifying first-order correlations of this feature space and neuronal behavior, but higher-order properties can be also explored in the future.

4.4 Conclusion

Just as in Chapter 3, we see data-driven deep learning can be applied to biological datasets to functionally learn class representations with minimal bias. Here, w-kNN voting of NPID-trained feature vectors yielded accuracy that compared to supervised models on pre-defined acoustical features [37], showing that these data-driven features are as salient as expert-defined features.

Additionally, NPID is also applied on a different domain of data: audio. We discover that representation learning generalizes to this domain also, though object structures do not easily correlate to those learned from vision.

Instead, MPS-defined receptive fields can functionally describe artificial neuronal behavior in neural netwrorks. Also, through comparisons of MTF and neuronal behavior projections onto orthogonally defined bases in the MPS feature space, some neurons' behavior can be functionally described by first or second order formulations of these bases. This project led to data-driven discovery of feature bases that are unique to biological audio representations.

Lastly, MPS-defined receptive fields also describe biological neuronal behavior in zebra finches, and the functionally calculated receptive fields match well between artificial and biological neurons. This commonality implies shared optimal learning of functionally-relevant representations from audio samples between both artificial and biological neural systems.

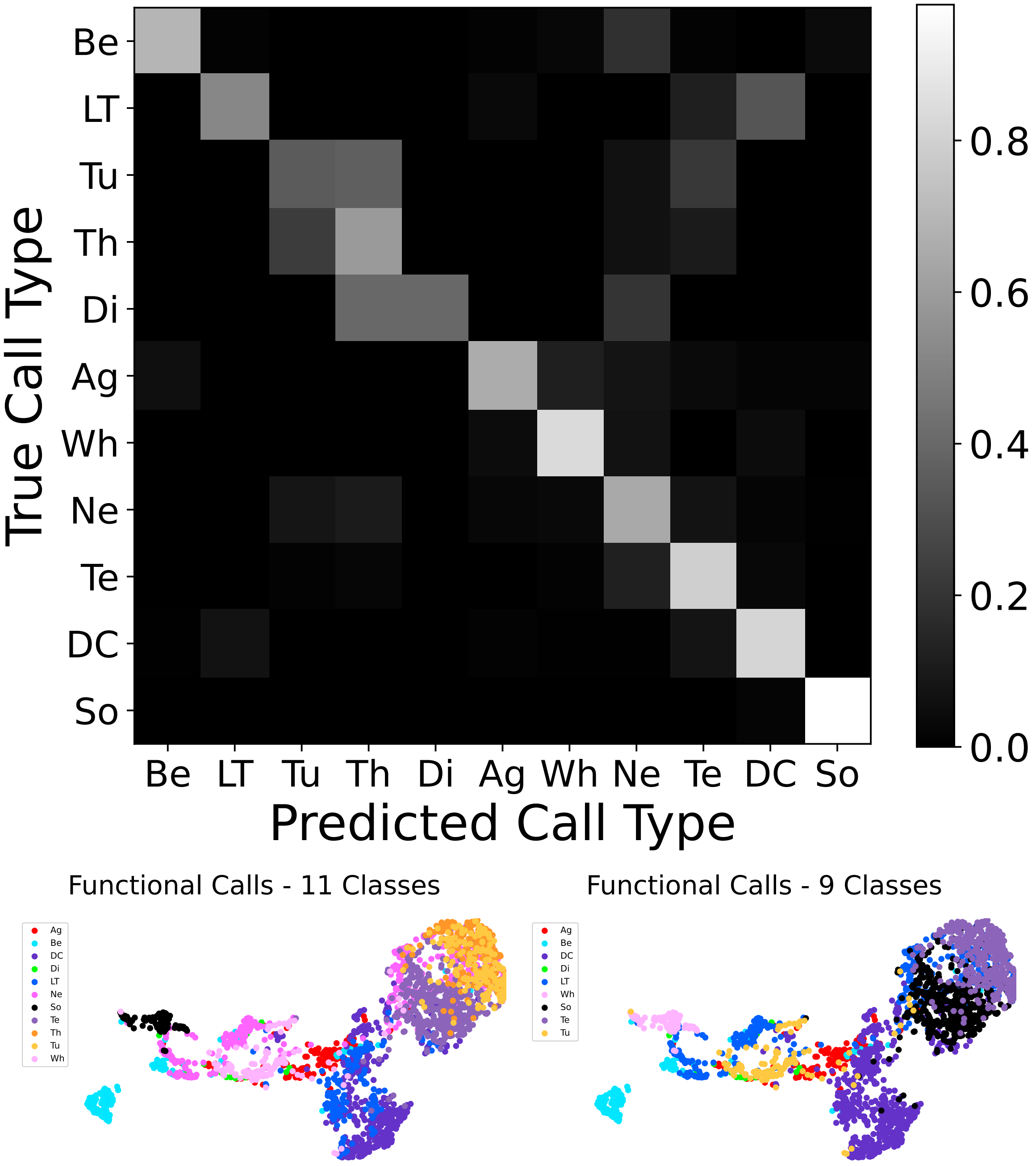


Figure 4.6: Confusion matrix and UMAP visualizations of best NPID-traned network performance. (Top) Confusion matrix showing class confusions. (Bottom) UMAP visualizations of the output training feature encodings describe underlying confusion matrix results. This visualization reveals which neighborhoods are distinct and which are mixed that can lead to confusions in the confusion matrix.

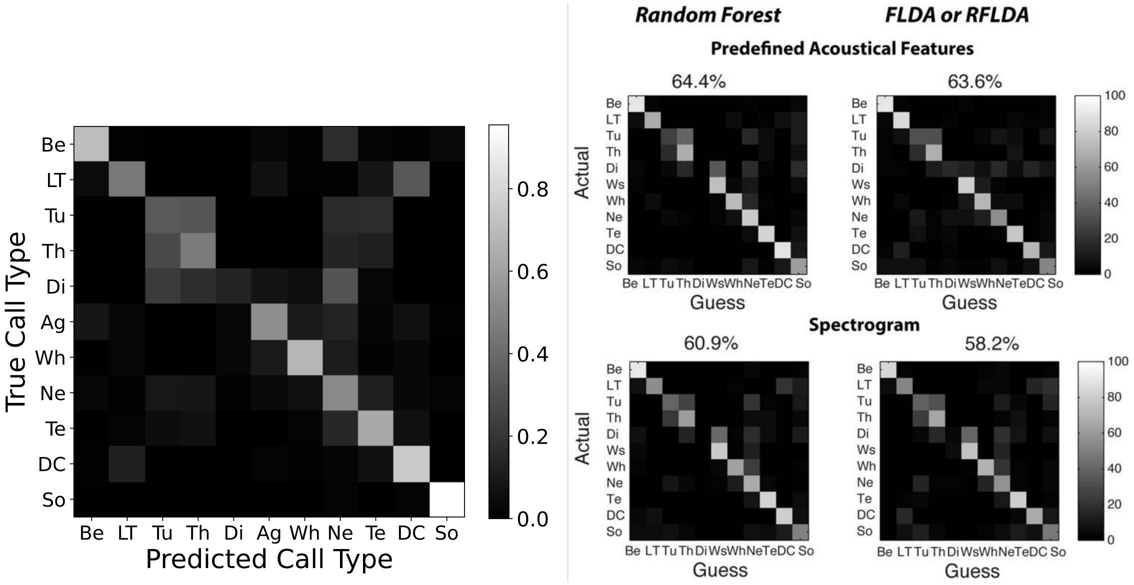


Figure 4.7: Posterior Probability Matrix. (Left) Posterior probability estimates derived from w-kNN voting normalized by a presumed uniform prior across classes. (Right) Posterior probability estimates derived from a published supervised baseline [37]. Our results are comparable to their spectrogram-based classifers.

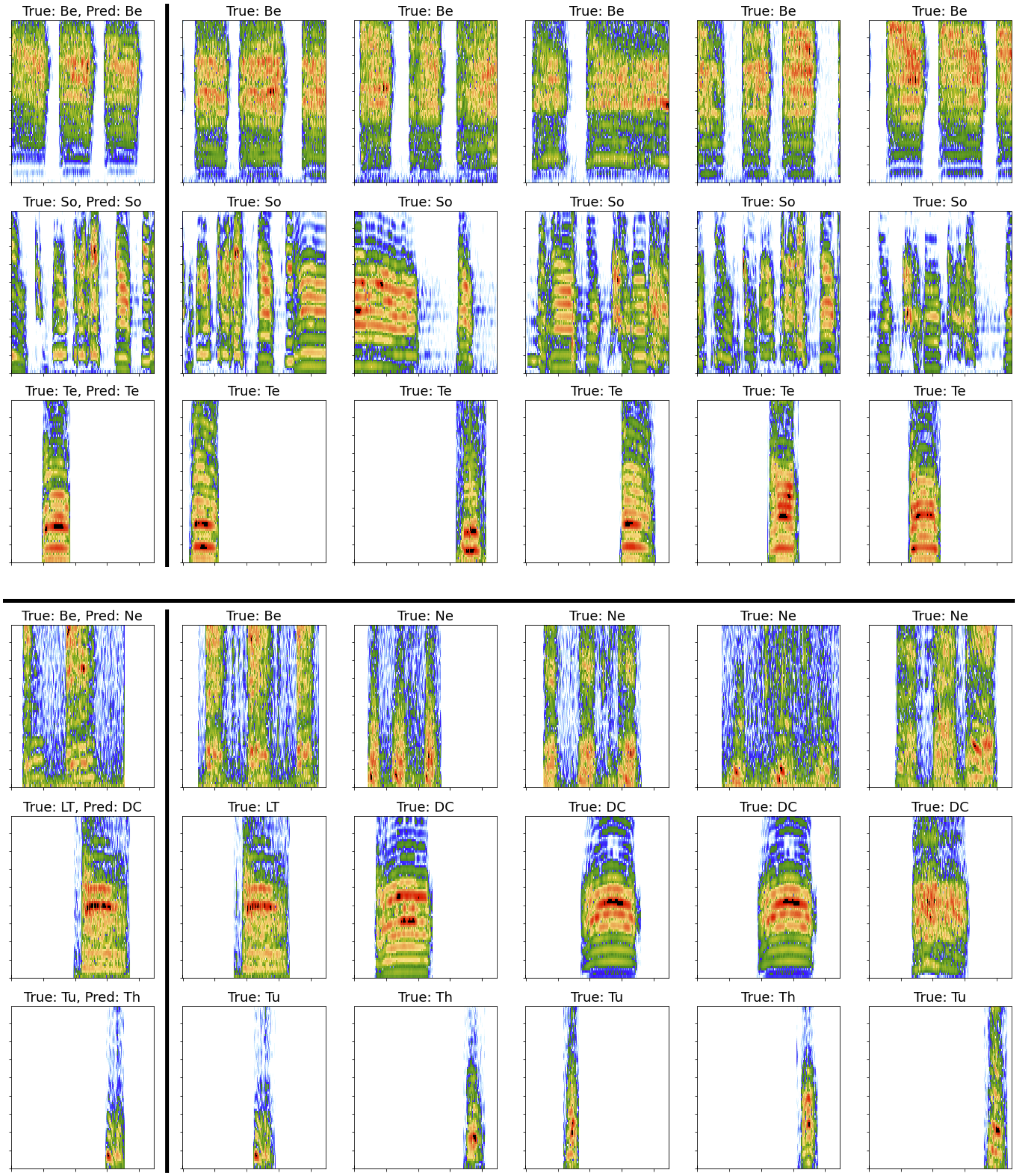


Figure 4.8: Retrievals with NPID. (Top) Retrievals leading to successful class predictions. (Bottom) Retrievals leading to failed class predictions. Leftmost column shows query images. Rightward columns show nearest 5 encoded training samples. There is heavy visual similarity between query and retrivals for both the successful and failed predictions, driving further investigation into why misclassified samples were confused with other classes.

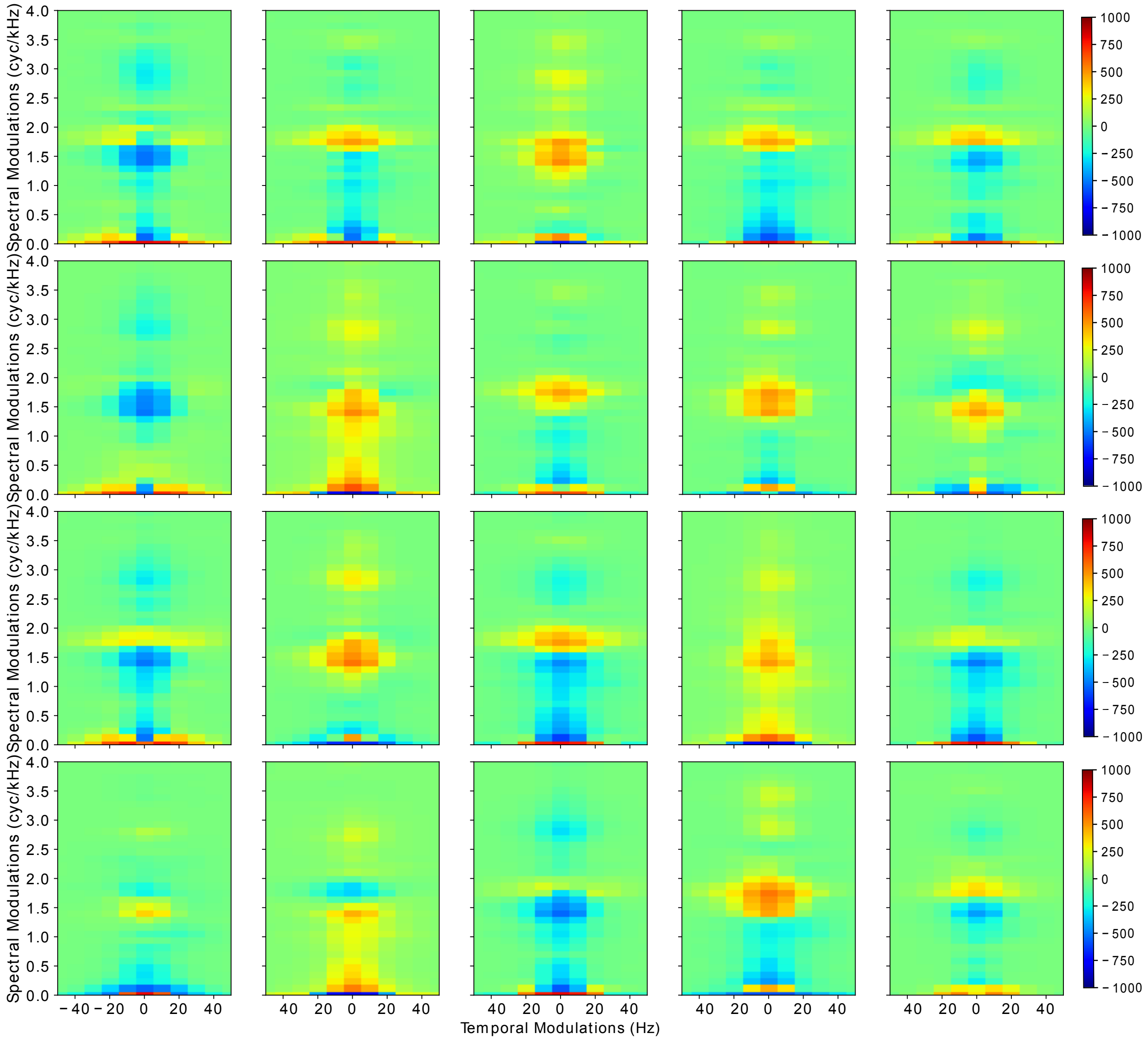


Figure 4.9: Artificial MTFs. Receptive fields generated in the MPS space from input sample MPS representations weighted by output CNN neuronal activations to those samples. The MTFs describe presumed neuronal preferences to mid-level audio features. We see receptive field relate to class-related harmonics, high-frequency, and low-frequency preferences.

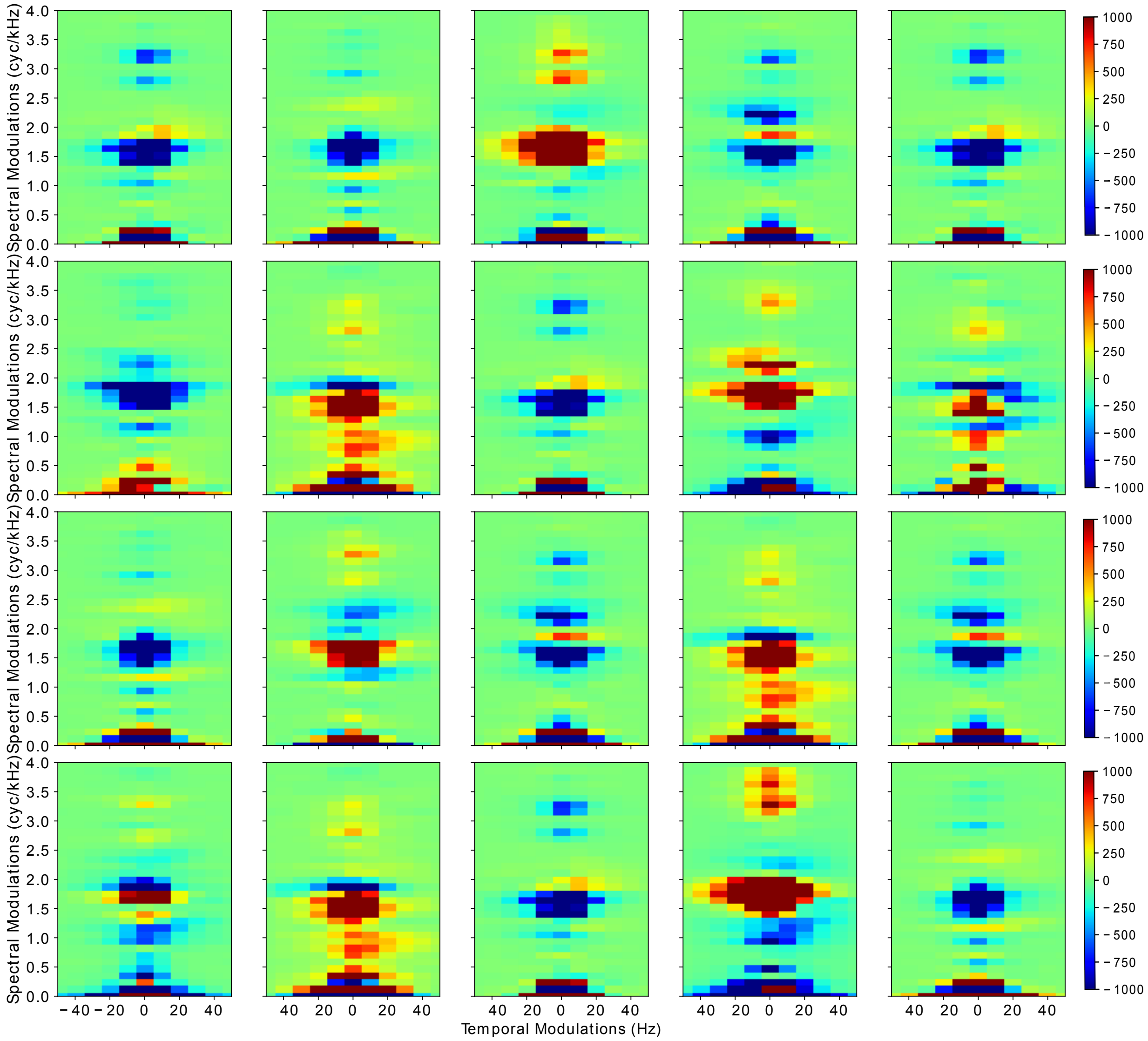


Figure 4.10: Real MRFs. Receptive fields generated in the MPS space from input sample MPS representations weighted by output zerba finch neuronal activations to those samples, sorted by highest cosine similarity to corresponding artificial MTFs from Fig 4.9

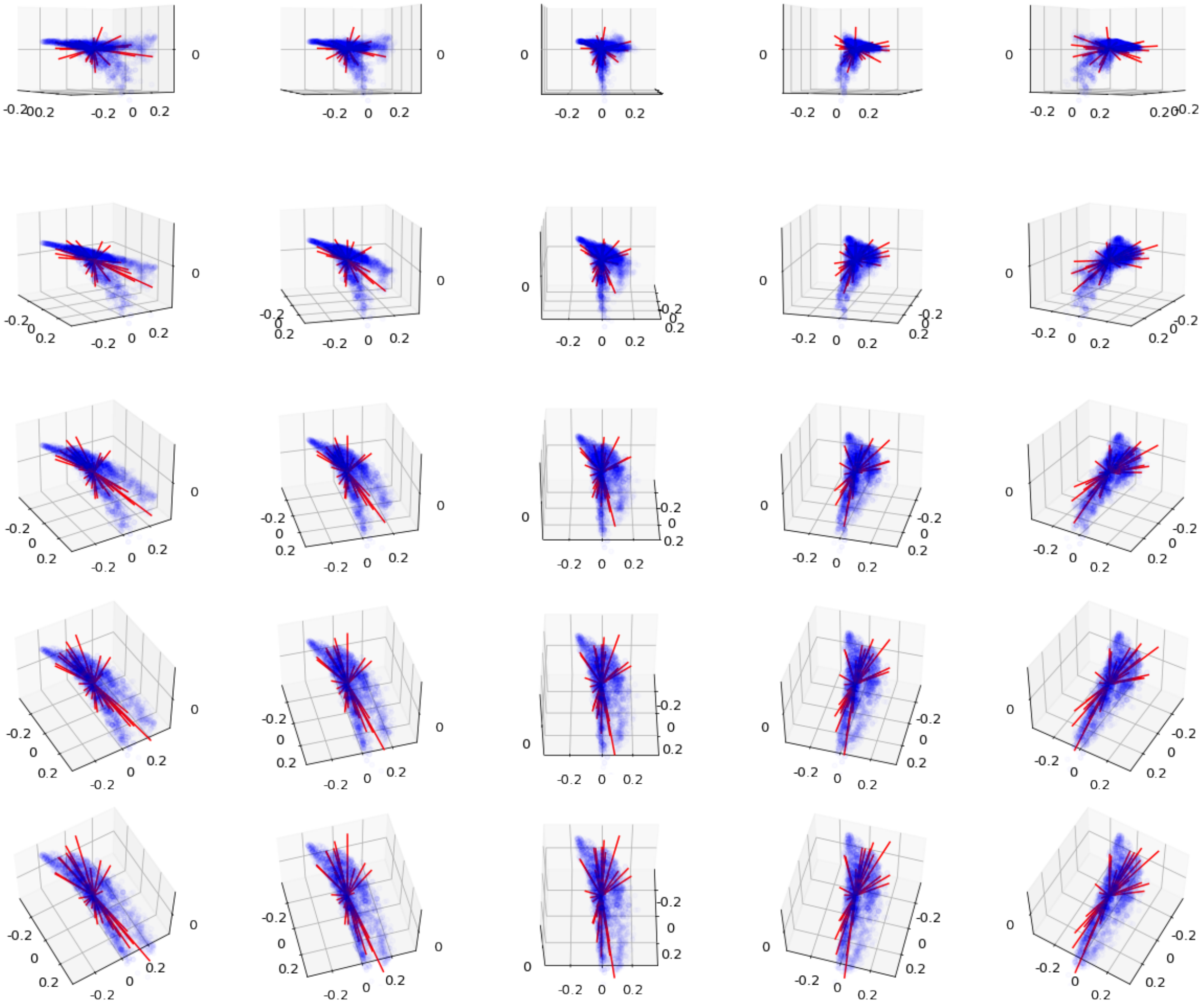


Figure 4.11: Artificial MTFs and Input MPSes projected onto Their PCs. With the top-3 PCs generated from the artificial MTFs as the bases, all MTFs and input MPSes are projected and plotted to assess salience of PC bases. There is significant overlap between the data distribution MPS features and presumed neural receptive field vectors in this feature space.

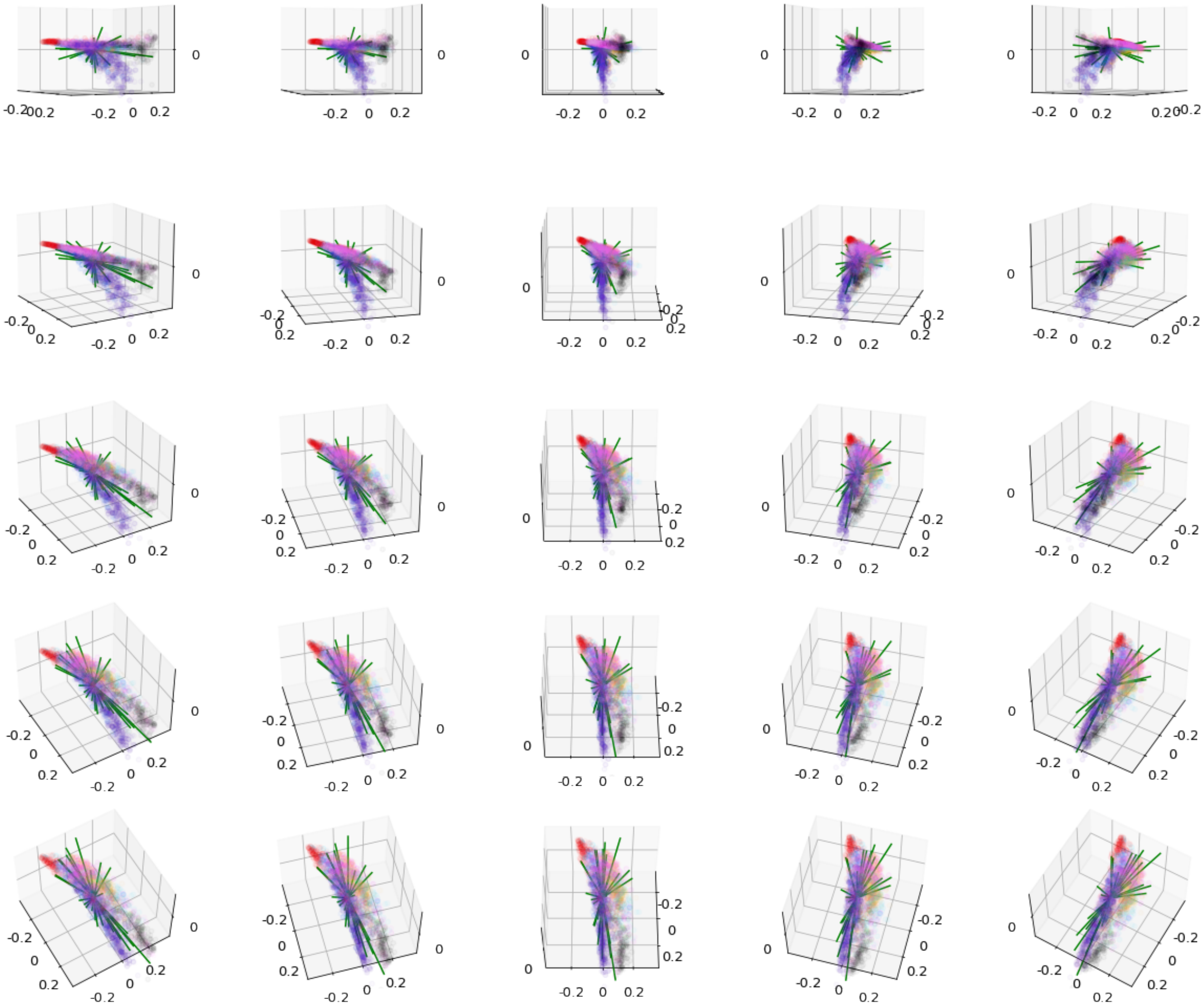


Figure 4.12: Artificial MTFs and Input MPSes projected onto Their PCs (colored by vocalization class). With the top-3 PCs generated from the artificial MTFs as the bases, the MTFs and input MPSes are projected and plotted to assess salience of PC bases. Input MPS coordinates are color-coded by their vocalization class as defned by the legend in Fig 4.6. There is class discriminability between the input sample MPS features projected onto these bases, implying MPS is a salient feature for the task.

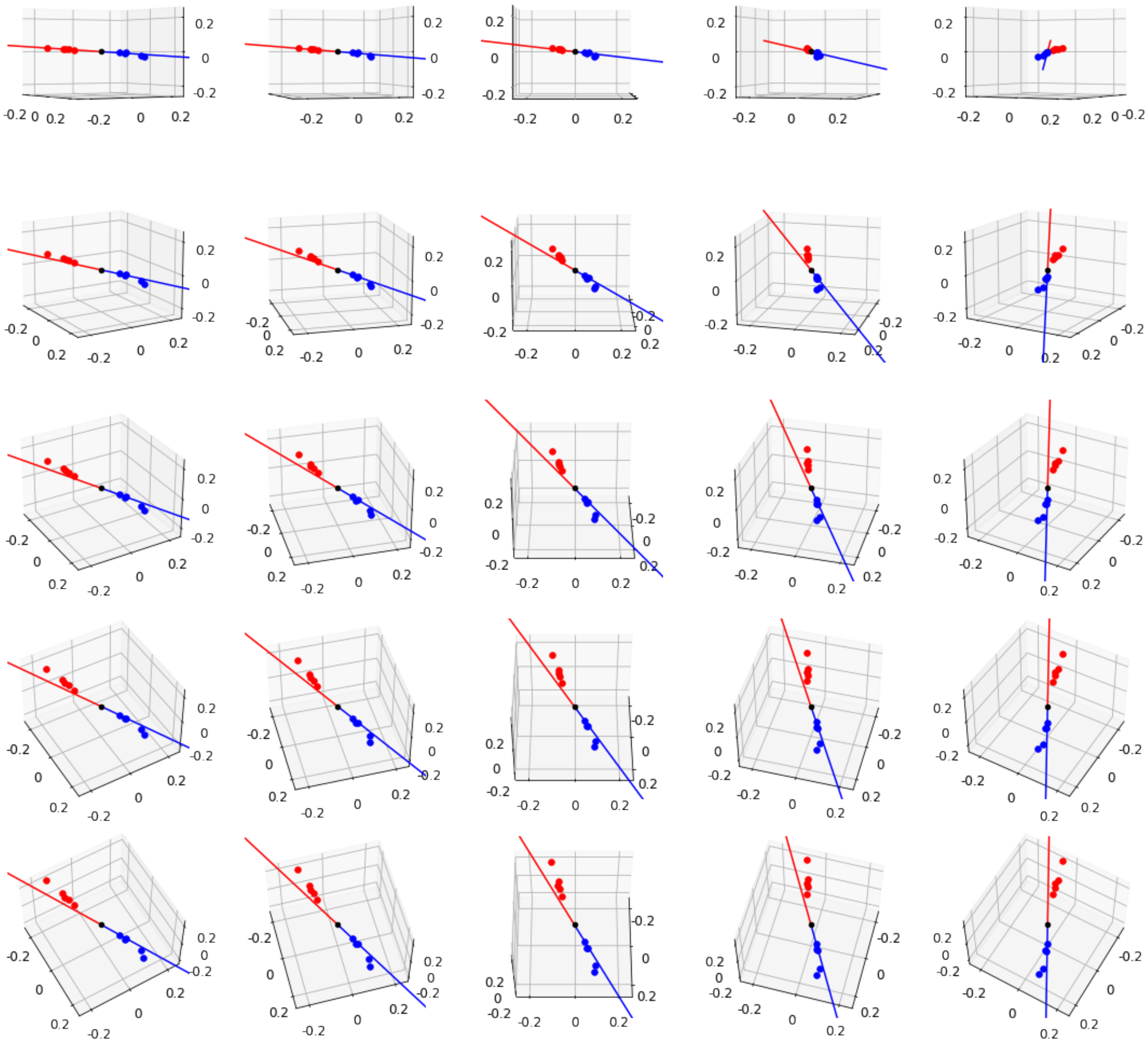


Figure 4.13: Example artificial MTF with linearly defined preferences. Line plot showing how MTF from output artificial Neuron 0 is projected onto the top 3 MTF PC bases (positive in red, negative in blue). Projections of the average MPS of the 5, 10, 20, 40, 100 min-activating (blue) and max-activating (red) samples for this neuron are plotted. This neuron's behavior can be described by a linear function of these PC bases.

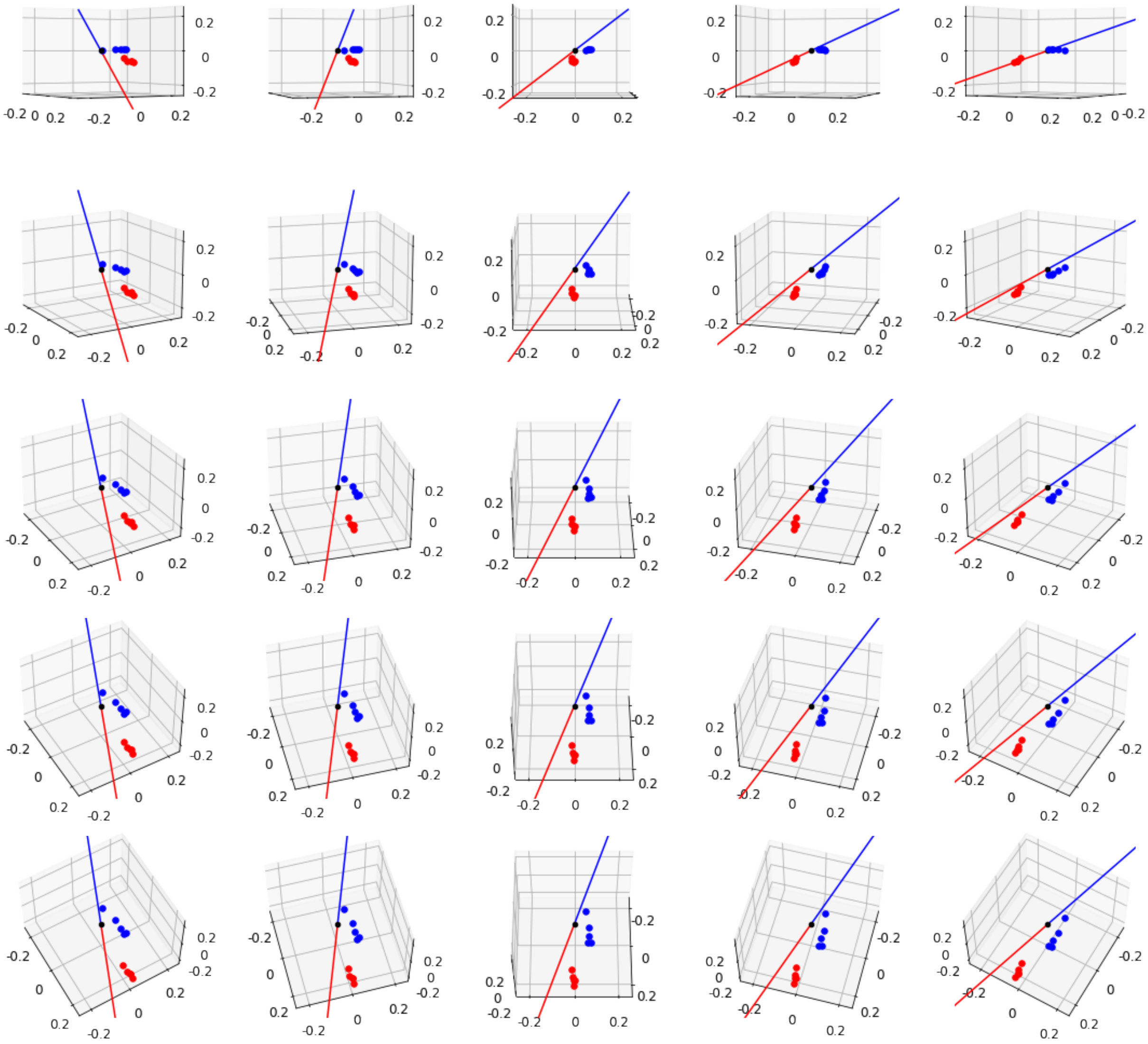


Figure 4.14: Example artificial MTF with quadratically defined preferences. Line plot showing how MTF from output artificial Neuron 6 is projected onto the top 3 MTF PC bases (positive in red, negative in blue). Projections of the average MPS of the 5, 10, 20, 40, 100 min-activating (blue) and max-activating (red) samples for this neuron are plotted. This neuron's behavior can be described by a quadratic function of these PC bases.

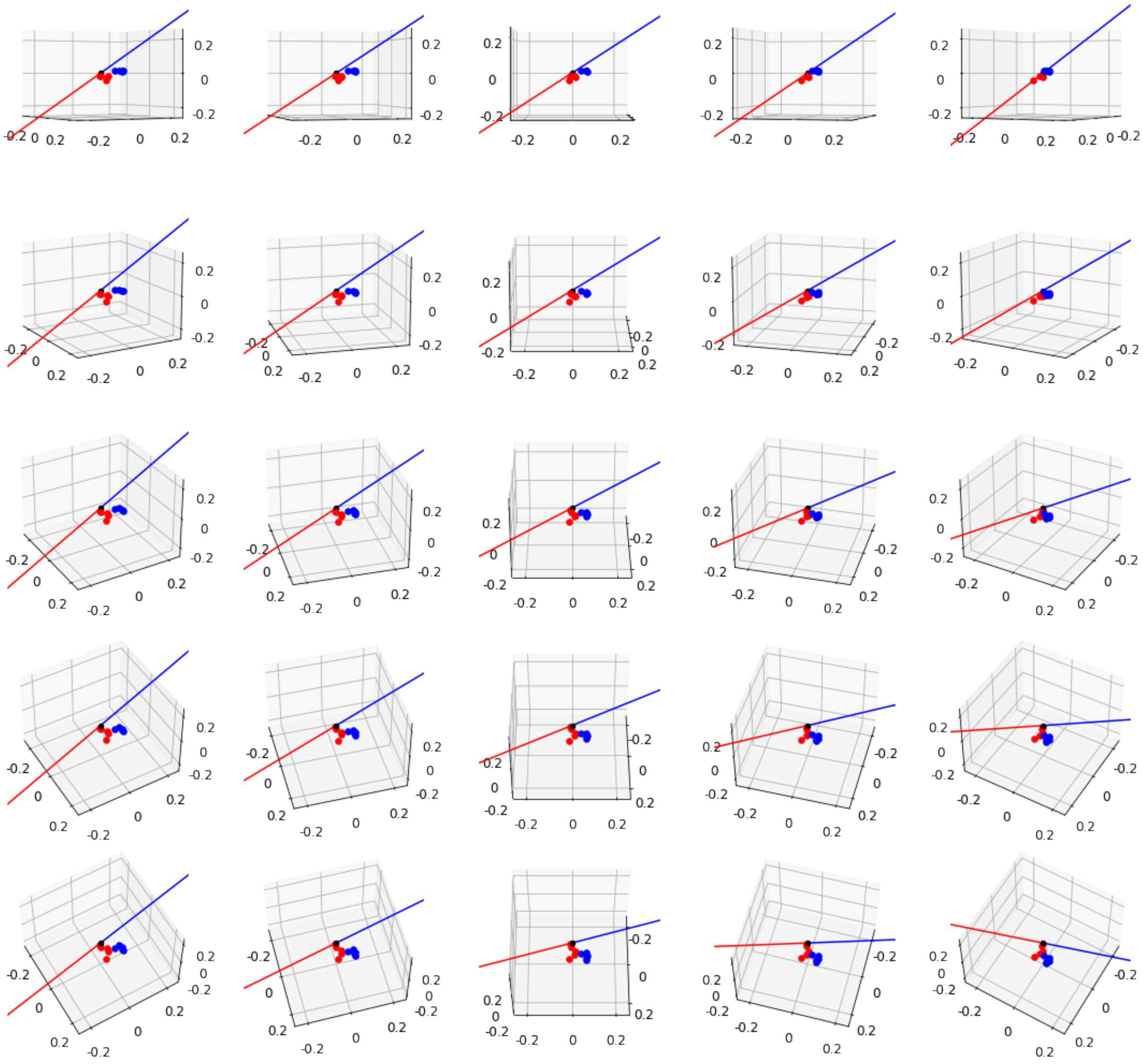


Figure 4.15: Example artificial MTF with higher-order preferences. Line plot showing how MTF from output artificial Neuron 3 is projected onto the top 3 MTF PC bases (positive in red, negative in blue). Projections of the average MPS of the 5, 10, 20, 40, 100 min-activating (blue) and max-activating (red) samples for this neuron are plotted. This neuron's behavior can be described by a higher order function of these PC bases.

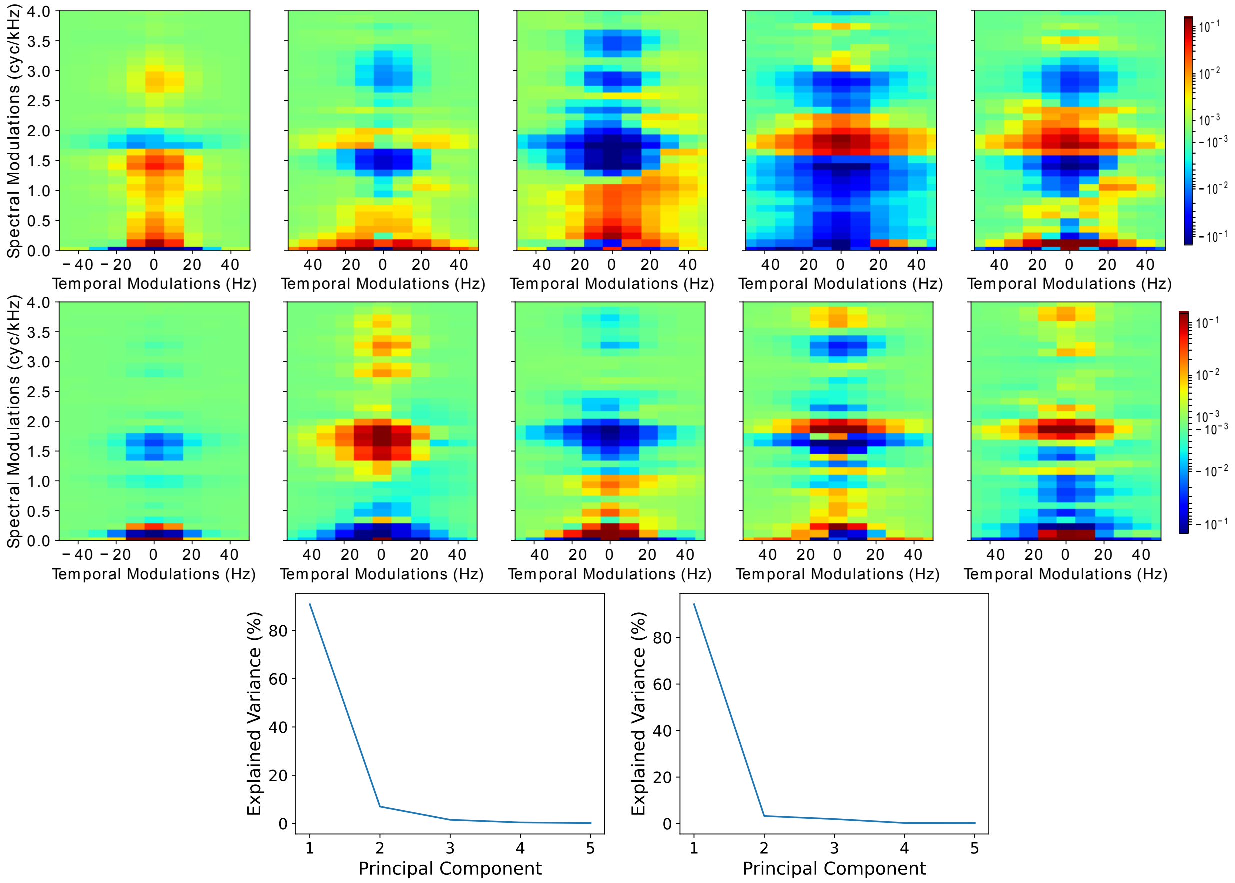


Figure 4.16: Artificial and Real MTF PCs. Comparison of the top 5 PCs generated from Artificial (top) and Real (bottom) MTFs. Corresponding explained variance plots are to the right of each PC plot. For both artifical and biological PCs, the top 3 PCs explain almost all variability and can be a useful set of bases for evaluation.

Chapter 5

Conclusion

This dissertation aims to take inspiration from biological neural modeling experiments to evaluate learned features from deep learning through data-driven discovery. Each project listed relates to feature exploration across a different domain of input data (self-driving video, medical image, and biological audio) to functionally describe learned features unique to each domain. The goal of this dissertation is to demonstrate that studying neural networks can reveal insights about what input representations are actually salient for accomplishing any underlying task.

Chapter 2 shows that we can take inspiration from neuroscience-style experiments to model artificial neuronal behavior. Here, a remote-controlled car with a CNN processing unit was trained to perform obstacle-avoidance through imitation learning in various structured and unstructured driving environments. Overall, the only labels for the data are motion-related, as steering and motor outputs were auto-generated from a joystick controller during data collection. Using these minimal labels, input videos were manipulated through framerate and frame order conditions and output activity was correlated to those conditions. Sensitivity plots for output neurons showed temporal dependence was learned differently between motor and steering tasks in same network processing the same time-series data, suggesting optical flow features mattered for steering prediction but not for motor prediction. The salience of these features can be assessed across the layers of the network, suggesting that framerate and frame order are correlated with early temporal representations in learned feature hierarchies. On a personal note, feature visualization learned on custom data and tasks can yield important insights, but it can be difficult to relate them to research on other data because there is no common point of evaluation; thus, performance results and verification of data viability should be of higher priority. Nevertheless, this project is the first evidence in this dissertation that data-driven analysis can yield insights into the underlying task.

In Chapter 3, data driven learning on medical retinal image data using NPID yields feature encodings that are functionally relevant to classification of AMD severity. Accuracy results were also comparable to published supervised and expert baselines. Though expert labels are needed for evaluation, learning occurs without them. Feature space boundaries are determined by the distribution of input training features, as opposed to human-defined boundaries, so this clinical bias is minimized for the classification task. Further cluster analysis of this learned feature space has revealed that globally organized neighborhoods are relevant for AMD classification, while locally organized neighborhoods are correlated with other clinically-relevant physiology and pathology. In addition, learning on hierarchically-organized subsets yields discovery of common patterns across mislabeled or easily confused data, providing insights about features that drive misclassification for AMD classification. Self-supervised learning on AREDS patient data revealed how useful feature interpretation is in the medical domain. Here, datasets are not curated for computer vision, so there is a need for visual breakdown of related phenotypes. These imply that data-driven deep learning and better cluster analysis tools could aid clinicians organize and interpret patient data in the future.

Chapter 4 4 extends this idea about data-driven learning to the audio domain. Here, NPID applied on zebra finch data yielded feature encodings that were functionally relevant for classifying vocalization calls, comparably performing to a published supervised baseline. Even though deep learning on audio data is primarily supervised and on species-level classification, this data driven approach was able to learn features salient to fine vocalization, a harder task. Our CNN's inputs may be 2D spectrogram images, but the learned representations are organized with temporal structures than representations learned from naturalistic videos. We assessed these structures with an audio-specific feature space: the MPS. We confirmed that our interpretation of neuronal activity in MPS feature space could functionally organize neuronal behavior relative to input samples. Furthermore, through novel comparisons of MPS-defined receptive fields for our artificial model and zebra finch neuronal spike data, we showed that CNNs discriminate audio data through learned representations that also help the brain discriminate audio. Similar to medically curated datasets, I discovered through this project that there is a need for representation visualization in (a) non-visual data domains, where learned representations can still be structured, and (b) biological neural data, where biological behavior needs to still be explained. Finally, common representations imply that similar encodings are optimally learned through these artificial and biological neural processing systems, so data-driven learning can assist neuroscientists compare learned representations with minimal bias.

Overall, this dissertation has evaluated deep learning applied on a host of real world tasks aside from standard datasets curated for computer vision: in-house self-driving video, medical retinal image, and zebra finch audio data. Though each project requires a different lens for explaining functionally salient behavior, we offer data-driven insights into each learned task that seem to be consistent with experimental findings in neuroscience and medicine.

Bibliography

[1] Michael David Abràmoff et al. “Improved automated detection of diabetic retinopathy on a publicly available dataset through integration of deep learning”. In: *Investigative Ophthalmology and Visual Science* 57.13 (2016). ISSN: 15525783. DOI: 10.1167/iovs.16-19964.

[2] Sharath Adavanne et al. “Stacked Convolutional and Recurrent Neural Networks for Bird Audio Detection”. In: (June 2017). URL: http://arxiv.org/abs/1706.02047.

[3] Haider Ali. “Hierarchical object classification using ImageNet domain ontologies”. In: *VISAPP 2010 - Proceedings of the International Conference on Computer Vision Theory and Applications*. Vol. 2. 2010, pp. 534–536. ISBN: 9789896740283. DOI: 10.5220/0002851905340536.

[4] AREDS Research Group. “The age-related eye disease study (AREDS) system for classifying cataracts from photographs: AREDS report no. 4 Members of the Age-Related Eye Disease Study Research Group are listed at the end of the article.” In: *American Journal of Ophthalmology* 131.2 (2001). ISSN: 00029394. DOI: 10.1016/s0002-9394(00)00732-7.

[5] AREDS Research Group. “The age-related eye disease study system for classifying age-related macular degeneration from stereoscopic color fundus photographs: The age-related eye disease study report number 6”. In: *American Journal of Ophthalmology* 132.5 (2001), pp. 668–681. ISSN: 00029394. DOI: 10.1016/S0002-9394(01)01218-1.

[6] Craig A Atencio, Tatyana O Sharpee, and Christoph E Schreiner. “Receptive field dimensionality increases from the auditory midbrain to cortex”. In: *Journal of Neurophysiology* 107.10 (2012), pp. 2594–2603. ISSN: 0022-3077.

[7] Joseph J. Atick and A. Norman Redlich. “What Does the Retina Know about Natural Scenes?” In: *Neural Computation* 4.2 (Mar. 1992), pp. 196–210. ISSN: 0899-7667. DOI: 10.1162/neco.1992.4.2.196.

[8] David Bau et al. “Network Dissection: Quantifying Interpretability of Deep Visual Representations”. In: (Apr. 2017). URL: http://arxiv.org/abs/1704.05796.

[9] Miguel A Bautista et al. *CliqueCNN: Deep Unsupervised Exemplar Learning*. Tech. rep. URL: https://github.com/asanakoy/cliquecnn.

[10] Robert M Bell and Yehuda Koren. *Lessons from the Netflix Prize Challenge*. Tech. rep.

[11] Y Bengio et al. *Visualizing Higher-Layer Features of a Deep Network Visualizing Higher-Layer Features of a Deep Network Département d'Informatique et Recherche Opérationnelle*. Tech. rep. 2009. URL: https://www.researchgate.net/publication/265022827.

[12] Peter C. Bermant et al. "Deep Machine Learning Techniques for the Detection and Classification of Sperm Whale Bioacoustics". In: *Scientific Reports* 9.1 (Dec. 2019). ISSN: 20452322. DOI: 10.1038/s41598-019-48909-4.

[13] Mariusz Bojarski et al. "End to End Learning for Self-Driving Cars". In: (2016), pp. 1–9. URL: http://arxiv.org/abs/1604.07316.

[14] Mariusz Bojarski et al. "Explaining How a Deep Neural Network Trained with End-to-End Learning Steers a Car". In: (Apr. 2017). URL: http://arxiv.org/abs/1704.07911.

[15] Mariusz Bojarski et al. "The NVIDIA PilotNet Experiments". In: (Oct. 2020). URL: http://arxiv.org/abs/2010.08776.

[16] J W Bradbury and S L WVehrencamp. *Animal Communication*. Sunderland, MA: Sinauer Associates, 1998.

[17] Andreas Burkhalterl and David C Van Essen. *Processing of Color, Form and Disparity Information in Visual Areas VP and V2 of Ventral Extrastriate Cortex in the Macaque Monkey*. Tech. rep. 1986, pp. 2327–2351.

[18] Philippe Burlina et al. "Comparing humans and deep learning performance for grading AMD: A study in using universal deep features and transfer learning for automated AMD analysis". In: *Computers in Biology and Medicine* 82 (2017). ISSN: 18790534. DOI: 10.1016/j.compbiomed.2017.01.018.

[19] Philippe M. Burlina et al. "Automated grading of age-related macular degeneration from color fundus images using deep convolutional neural networks". In: *JAMA Ophthalmology* 135.11 (2017). ISSN: 21686165. DOI: 10.1001/jamaophthalmol.2017.3782.

[20] Velia Cardin and Andrew T. Smith. "Sensitivity of Human Visual and Vestibular Cortical Regions to Egomotion-Compatible Visual Stimulation". In: *Cerebral Cortex* 20.8 (Aug. 2010), pp. 1964–1973. ISSN: 1460-2199. DOI: 10.1093/cercor/bhp268.

[21] N L Carlson, V L Ming, and M R Deweese. "Sparse codes for speech predict spectrotemporal receptive fields in the inferior colliculus". In: *PLoS Comput Biol* 8.7 (2012), e1002594. ISSN: 1553-7358 (Electronic)1553-734X (Linking). DOI: 10.1371/journal.pcbi.1002594. URL: http://www.ncbi.nlm.nih.gov/pubmed/22807665http://www.ncbi.nlm.nih.gov/pmc/articles/PMC3395612/pdf/pcbi.1002594.pdf.

[22] James R. Cavanaugh, Wyeth Bair, and J. Anthony Movshon. "Nature and interaction of signals from the receptive field center and surround in macaque V1 neurons". In: *Journal of Neurophysiology* 88.5 (Nov. 2002), pp. 2530–2546. ISSN: 00223077. DOI: 10.1152/jn.00692.2001.

[23] Boyuan Chen et al. "The Boombox: Visual Reconstruction from Acoustic Vibrations". In: *arXiv preprint arXiv:2105.08052* (2021).

[24] Changan Chen, Ziad Al-Halah, and Kristen Grauman. "Semantic Audio-Visual Navigation". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2021, pp. 15516–15525.

[25] Changan Chen et al. "Soundspaces: Audio-visual navigation in 3d environments". In: *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part VI 16*. 2020, pp. 17–36.

[26] Ting Chen et al. *A simple framework for contrastive learning of visual representations*. 2020.

[27] Xi Chen et al. "InfoGAN: Interpretable Representation Learning by Information Maximizing Generative Adversarial Nets". In: *arXiv* (2016). URL: http://arxiv.org/abs/1606.03657.

[28] D L Cheney and R M Seyfarth. "Assessment of Meaning and the Detection of Unreliable Signals by Vervet Monkeys". In: *Animal Behaviour* 36 (1988), pp. 477–486. ISSN: 0003-3472.

[29] Jesper Haahr Christensen, Sascha Hornauer, and Stella Yu. "BatVision: Learning to See 3D Spatial Layout with Two Ears". In: (Dec. 2019). URL: http://arxiv.org/abs/1912.07011.

[30] Mark Christopher et al. "Performance of Deep Learning Architectures and Transfer Learning for Detecting Glaucomatous Optic Neuropathy in Fundus Photographs". In: *Scientific Reports* 8.1 (2018). ISSN: 20452322. DOI: 10.1038/s41598-018-35044-9.

[31] Dena J. Clink and Holger Klinck. "Unsupervised acoustic classification of individual gibbon females and the implications for passive acoustic monitoring". In: *Methods in Ecology and Evolution* 12.2 (Feb. 2021), pp. 328–341. ISSN: 2041-210X. DOI: 10.1111/2041-210X.13520.

[32] N E Collias. "THE VOCAL REPERTOIRE OF THE RED JUNGLEFOWL - A SPECTROGRAPHIC CLASSIFICATION AND THE CODE OF COMMUNICATION". In: *Condor* 89.3 (1987), pp. 510–524. ISSN: 0010-5422. DOI: 10.2307/1368641. URL: <GotoISI>://WOS:A1987J821100007http://www.jstor.org/stable/pdfplus/1368641.pdf?acceptTC=truehttp://www.jstor.org/stable/10.2307/1368641?origin=crossref.

[33] Bevil R. Conway, Sebastian Moeller, and Doris Y. Tsao. "Specialized Color Modules in Macaque Extrastriate Cortex". In: *Neuron* 56.3 (Nov. 2007), pp. 560–573. ISSN: 08966273. DOI: 10.1016/j.neuron.2007.10.008.

[34] Matthew D. Davis et al. "The age-related eye disease study severity scale for age-related macular degeneration: AREDS report no. 17". In: *Archives of Ophthalmology* 123.11 (2005). ISSN: 00039950. DOI: 10.1001/archopht.123.11.1484.

[35] Alexey Dosovitskiy et al. *CARLA: An Open Urban Driving Simulator*. Tech. rep.

[36] Alexey Dosovitskiy et al. "Discriminative unsupervised feature learning with exemplar convolutional neural networks". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 38.9 (2016). ISSN: 01628828. DOI: 10.1109/TPAMI.2015.2496141.

[37] J E Elie and F E Theunissen. "The vocal repertoire of the domesticated zebra finch: a data-driven approach to decipher the information-bearing acoustic features of communication signals". In: *Animal Cognition* 19.2 (2016), pp. 285–315. ISSN: 1435-9448. DOI: 10.1007/s10071-015-0933-6. URL: <GotoISI>://WOS:000370170300004https://www.ncbi.nlm.nih.gov/pmc/articles/PMC5973879/pdf/nihms966595.pdf.

[38] Julie E Elie and Frédéric E Theunissen. "Zebra finches identify individuals using vocal signatures unique to each call type". In: *Nature Communications* 9.1 (2018), p. 4026. ISSN: 2041-1723. DOI: 10.1038/s41467-018-06394-9. URL: https://doi.org/10.1038/s41467-018-06394-9.

[39] Haoqi Fan et al. "Multiscale Vision Transformers". In: (Apr. 2021). URL: http://arxiv.org/abs/2104.11227.

[40] Frederick L. Ferris et al. "A simplified severity scale for age-related macular degeneration: AREDS report no. 18". In: *Archives of Ophthalmology* 123.11 (2005). ISSN: 00039950. DOI: 10.1001/archopht.123.11.1570.

[41] M S Ficken, R W Ficken, and S R Witkin. "VOCAL REPERTOIRE OF BLACK-CAPPED CHICKADEE". In: *Auk* 95.1 (1978), pp. 34–48. ISSN: 0004-8038. URL: <GotoISI>://WOS:A1978EL69900004.

[42] J Fiser et al. "Statistically optimal perception and learning: from behavior to neural representations". In: *Trends in Cognitive Sciences* 14.3 (2010), pp. 119–130. ISSN: 1364-6613. DOI: 10.1016/j.tics.2010.01.003. URL: <GotoISI>://WOS:000275605900004.

[43] Jack L Gallant, Rachel E Shoup, and James A Mazer. "A Human Extrastriate Area Functionally Homologous to Macaque V4". In: *Neuron* 27.2 (Aug. 2000), pp. 227–235. ISSN: 08966273. DOI: 10.1016/S0896-6273(00)00032-5.

[44] Jack L Gallant et al. *Neural Responses to Polar, Hyperbolic, and Cartesian Gratings in Area V4 of the Macaque Monkey*. Tech. rep. 4. 1996.

[45] Chuang Gan et al. "Self-supervised moving vehicle tracking with stereo sound". In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 2019, pp. 7053–7062.

[46] Ruohan Gao and Kristen Grauman. "Visualvoice: Audio-visual speech separation with cross-modal consistency". In: *arXiv preprint arXiv:2101.03149* (2021).

[47] Ruohan Gao et al. “Visualechoes: Spatial image representation learning through echolocation”. In: *European Conference on Computer Vision*. 2020, pp. 658–676.

[48] M Garcia et al. “Evolution of communication signals and information during species radiation”. In: *Nature Communications* 11.1 (2020). ISSN: 2041-1723. DOI: 10.1038/s41467-020-18772-3. URL: <GotoISI>://WOS:000577122600001.

[49] Rishab Gargeya and Theodore Leng. “Automated Identification of Diabetic Retinopathy Using Deep Learning”. In: *Ophthalmology* 124.7 (2017). ISSN: 15494713. DOI: 10.1016/j.ophtha.2017.02.008.

[50] Leon A. Gatys, Alexander S. Ecker, and Matthias Bethge. “Texture Synthesis Using Convolutional Neural Networks”. In: (May 2015). URL: http://arxiv.org/abs/1505.07376.

[51] K. R. Gegenfurtner, D. C. Kiper, and S. B. Fenstemaker. “Processing of color, form, and motion in macaque area V2”. In: *Visual Neuroscience* 13.1 (1996), pp. 161–172. ISSN: 09525238. DOI: 10.1017/S0952523800007203.

[52] Ross Girshick. “Fast R-CNN”. In: (Apr. 2015). URL: http://arxiv.org/abs/1504.08083.

[53] Ij Goodfellow, J Pouget-Abadie, and Mehdi Mirza. “Generative Adversarial Networks”. In: *arXiv preprint arXiv: ...* (2014), pp. 1–9. ISSN: 10495258. URL: http://arxiv.org/abs/1406.2661.

[54] Felix Grassmann et al. “A Deep Learning Algorithm for Prediction of Age-Related Eye Disease Study Severity Scale for Age-Related Macular Degeneration from Color Fundus Photography”. In: *Ophthalmology* 125.9 (2018). ISSN: 15494713. DOI: 10.1016/j.ophtha.2018.02.037.

[55] P A Green, N C Brandley, and S Nowicki. “Categorical perception in animal communication and decision-making”. In: *Behavioral Ecology* 31.4 (2020), pp. 859–867. ISSN: 1045-2249. DOI: 10.1093/beheco/araa004. URL: <GotoISI>://WOS:000591672200001.

[56] Varun Gulshan et al. “Development and validation of a deep learning algorithm for detection of diabetic retinopathy in retinal fundus photographs”. In: *JAMA - Journal of the American Medical Association* 316.22 (2016). ISSN: 15383598. DOI: 10.1001/jama.2016.17216.

[57] Bharath Hariharan et al. “Simultaneous Detection and Segmentation”. In: (July 2014). URL: http://arxiv.org/abs/1407.1808.

[58] Zhenliang He et al. “AttGAN: Facial Attribute Editing by Only Changing What You Want”. In: (Nov. 2017). URL: http://arxiv.org/abs/1711.10678.

[59] Alam Ahmad Hidayat, Tjeng Wawan Cenggoro, and Bens Pardamean. “Convolutional Neural Networks for Scops Owl Sound Classification”. In: *Procedia Computer Science* 179 (2021), pp. 81–87. ISSN: 18770509. DOI: 10.1016/j.procs.2020.12.010.

[60] R Devon Hjelm et al. "Learning deep representations by mutual information estimation and maximization". In: (Aug. 2018). URL: http://arxiv.org/abs/1808.06670.

[61] Yiqi Hou, Sascha Hornauer, and Karl Zipser. "Fast Recurrent Fully Convolutional Networks for Direct Perception in Autonomous Driving". In: (Nov. 2017). URL: http://arxiv.org/abs/1711.06459.

[62] John Houston et al. "One Thousand and One Hours: Self-driving Motion Prediction Dataset". In: (June 2020). URL: http://arxiv.org/abs/2006.14480.

[63] A Hsu et al. "Modulation power and phase spectrum of natural sounds enhance neural encoding performed by single auditory neurons". In: *J Neurosci* 24.41 (2004), pp. 9201–9211. ISSN: 1529-2401 (Electronic)0270-6474 (Linking). DOI: 10.1523/JNEUROSCI.2449-04.2004. URL: http://www.ncbi.nlm.nih.gov/pubmed/15483139.

[64] Forrest N. Iandola et al. "SqueezeNet: AlexNet-level accuracy with 50x fewer parameters and ¡0.5MB model size". In: (Feb. 2016). URL: http://arxiv.org/abs/1602.07360.

[65] Forrest N. Iandola et al. "SqueezeNet: AlexNet-level accuracy with 50x fewer parameters and ¡0.5MB model size". In: (Feb. 2016). URL: http://arxiv.org/abs/1602.07360.

[66] Laurent Itti, Christof Koch, and Ernst Niebur. *A Model of Saliency-based Visual Attention for Rapid Scene Analysis*. Tech. rep.

[67] Stefan Kahl et al. "BirdNET: A deep learning solution for avian diversity monitoring". In: *Ecological Informatics* 61 (Mar. 2021), p. 101236. ISSN: 15749541. DOI: 10.1016/j.ecoinf.2021.101236.

[68] Evangelos Kazakos et al. "Slow-Fast Auditory Streams for Audio Recognition". In: *ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. 2021, pp. 855–859.

[69] Alexander Kirillov et al. *Panoptic Segmentation*. Tech. rep. URL: https://arxiv.org/abs/1801.00868..

[70] A S Kozlov and T Q Gentner. "Central auditory neurons have composite receptive fields". In: *Proceedings of the National Academy of Sciences of the United States of America* 113.5 (2016), pp. 1441–1446. ISSN: 0027-8424. DOI: 10.1073/pnas.1506903113. URL: <GotoISI>://WOS:000369085100087.

[71] Jonathan Krause et al. "Grader Variability and the Importance of Reference Standards for Evaluating Machine Learning Models for Diabetic Retinopathy". In: *Ophthalmology* 125.8 (2018). ISSN: 15494713. DOI: 10.1016/j.ophtha.2018.01.034.

[72] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. *ImageNet Classification with Deep Convolutional Neural Networks*. Tech. rep. URL: http://code.google.com/p/cuda-convnet/.

[73] H Kruuk. *The Spotted Hyena. A study of predation and social behavior.* Univ. Chicago Press, 1972.

[74] P K Kuhl. “Early language acquisition: Cracking the speech code”. In: *Nature Reviews Neuroscience* 5.11 (2004), pp. 831–843. ISSN: 1471-003X. DOI: `10.1038/nrn1533`. URL: `<GotoISI>://WOS:000224785500013`.

[75] Hugo Larochelle et al. *Exploring Strategies for Training Deep Neural Networks Pascal Lamblin*. Tech. rep. 2009, pp. 1–40.

[76] Yuanqing Lin et al. “Imagenet classification: fast descriptor coding and large-scale svm training”. In: *Large scale visual recognition challenge* (2010).

[77] Geert Litjens et al. *A survey on deep learning in medical image analysis*. Dec. 2017. DOI: `10.1016/j.media.2017.07.005`.

[78] Sidong Liu et al. “A Deep Learning-Based Algorithm Identifies Glaucomatous Discs Using Monoscopic Fundus Photographs”. In: *Ophthalmology. Glaucoma* 1.1 (2018). ISSN: 25894196. DOI: `10.1016/j.ogla.2018.04.002`.

[79] R. Lyon. “A computational model of filtering, detection, and compression in the cochlea”. In: *ICASSP ’82. IEEE International Conference on Acoustics, Speech, and Signal Processing*. Institute of Electrical and Electronics Engineers, pp. 1282–1285. DOI: `10.1109/ICASSP.1982.1171644`.

[80] P Marler. “The voice of the chaffinch and its function as a language”. In: *Ibis* 98 (1956), pp. 231–261.

[81] Mark Martinez et al. “Beyond Grand Theft Auto V for Training, Testing and Enhancing Deep Learning in Self Driving Cars”. In: (Dec. 2017). URL: `http://arxiv.org/abs/1712.01397`.

[82] Simone Meyer et al. “Phase-based frame interpolation for video”. In: IEEE, June 2015, pp. 1410–1418. ISBN: 978-1-4673-6964-0. DOI: `10.1109/CVPR.2015.7298747`.

[83] Michael Montemerlo et al. *Winning the DARPA Grand Challenge with an AI Robot*. Tech. rep. URL: `www.aaai.org`.

[84] R Channing Moore, Tyler Lee, and Frederic E Theunissen. “Noise-invariant Neurons in the Avian Auditory Cortex: Hearing the Song in Noise”. In: *Plos Computational Biology* 9.3 (2013), e1002942. ISSN: 1553-7358.

[85] Veronica Morfi, Robert F. Lachlan, and Dan Stowell. “Deep perceptual embeddings for unlabelled animal sound events”. In: *The Journal of the Acoustical Society of America* 150.1 (July 2021), pp. 2–11. ISSN: 0001-4966. DOI: `10.1121/10.0005475`.

[86] S C Mouterde et al. “Acoustic communication and sound degradation: how do the individual signatures of male and female zebra finch calls transmit over distance?” In: *PLoS One* 9 (7 2014), e102842. ISSN: 1932-6203 (Electronic) 1932-6203 (Linking). DOI: `10.1371/journal.pone.0102842`. URL: `http://www.ncbi.nlm.nih.gov/pubmed/25061795`.

[87] Anirvan S. Nandy et al. "The Fine Structure of Shape Tuning in Area V4". In: *Neuron* 78.6 (June 2013), pp. 1102–1115. ISSN: 08966273. DOI: `10.1016/j.neuron.2013.04.016`.

[88] Anh Nguyen, Jason Yosinski, and Jeff Clune. "Understanding Neural Networks via Feature Visualization: A survey". In: (Apr. 2019). URL: `http://arxiv.org/abs/1904.08939`.

[89] Emma Ozanich et al. "Deep embedded clustering of coral reef bioacoustics". In: (Dec. 2020). DOI: `10.1121/10.0004221`. URL: `http://arxiv.org/abs/2012.09982http://dx.doi.org/10.1121/10.0004221`.

[90] Daniel S. Park et al. "SpecAugment: A Simple Data Augmentation Method for Automatic Speech Recognition". In: (Apr. 2019). DOI: `10.21437/Interspeech.2019-2680`. URL: `http://arxiv.org/abs/1904.08779http://dx.doi.org/10.21437/Interspeech.2019-2680`.

[91] Anitha Pasupathy and Charles E. Connor. "Population coding of shape in area V4". In: *Nature Neuroscience* 5.12 (Dec. 2002), pp. 1332–1338. ISSN: 10976256. DOI: `10.1038/nn972`.

[92] Deepak Pathak et al. *Context Encoders: Feature Learning by Inpainting.* Tech. rep.

[93] Yifan Peng et al. "DeepSeeNet: A Deep Learning Model for Automated Classification of Patient-based Age-related Macular Degeneration Severity from Color Fundus Photographs". In: *Ophthalmology* 126.4 (2019). ISSN: 15494713. DOI: `10.1016/j.ophtha.2018.11.015`.

[94] Emilie C Perez et al. "Physiological resonance between mates through calls as possible evidence of empathic processes in songbirds". In: *Hormones and behavior* 75 (2015), pp. 130–141. DOI: `10.1016/j.yhbeh.2015.09.002`. URL: `<GotoWoS>://WOS:000397823700007`.

[95] Florent Perronnin, Jorge Sanchez, and Thomas Mensink. *Improving the Fisher Kernel for Large-Scale Image Classification.* Tech. rep. URL: `http://www.image-net.org`.

[96] Archontis Politis et al. "Overview and evaluation of sound event localization and detection in DCASE 2019". In: *IEEE/ACM Transactions on Audio, Speech, and Language Processing* 29 (2020), pp. 684–698.

[97] Dean A Pomerleau. *ALVINN: AN AUTONOMOUS LAND VEHICLE IN A NEURAL NETWORK.* Tech. rep.

[98] Ryan Poplin et al. "Prediction of cardiovascular risk factors from retinal fundus photographs via deep learning". In: *Nature Biomedical Engineering* 2.3 (2018). ISSN: 2157846X. DOI: `10.1038/s41551-018-0195-0`.

[99] "portilla99-reprint". In: ().

[100] Senthil Purushwalkam et al. "Audio-Visual Floorplan Reconstruction". In: (Dec. 2020). URL: `http://arxiv.org/abs/2012.15470`.

[101] Jeppe Have Rasmussen and Ana Širović. “Automatic detection and classification of baleen whale social calls using convolutional neural networks”. In: *The Journal of the Acoustical Society of America* 149.5 (May 2021), pp. 3635–3644. ISSN: 0001-4966. DOI: `10.1121/10.0005047`.

[102] Zhihang Ren, Stella X. Yu, and David Whitney. “Controllable Medical Image Generation via Generative Adversarial Networks”. In: *Electronic Imaging* 2021.11 (Feb. 2021), pp. 112–1. ISSN: 2470-1173. DOI: `10.2352/issn.2470-1173.2021.11.hvei-112`.

[103] F A Rodriguez et al. “Neural modulation tuning characteristics scale to efficiently encode natural sound statistics”. In: *J Neurosci* 30.47 (2010), pp. 15969–15980. ISSN: 1529-2401 (Electronic)0270-6474 (Linking). DOI: `10.1523/JNEUROSCI.0966-10.2010`. URL: `http://www.ncbi.nlm.nih.gov/pubmed/21106835http://www.ncbi.nlm.nih.gov/pmc/articles/PMC3351116/pdf/nihms254151.pdf`.

[104] Bruno Rossion, Chun-Chia Kung, and Michael J Tarr. *Visual expertise with nonface objects leads to competition with the early perceptual processing of faces in the human occipitotemporal cortex.* Tech. rep. 2004, p. 2021. URL: `www.pnas.orgcgidoi10.1073pnas.0405613101`.

[105] Daniel L. Ruderman. “The statistics of natural images”. In: *Network: Computation in Neural Systems* 5.4 (1994). ISSN: 0954898X. DOI: `10.1088/0954-898X{\_}5{\_}4{\_}006`.

[106] Olga Russakovsky et al. “ImageNet Large Scale Visual Recognition Challenge”. In: *International Journal of Computer Vision* 115.3 (2015). ISSN: 15731405. DOI: `10.1007/s11263-015-0816-y`.

[107] J. A. Saunders. “View rotation is used to perceive path curvature from optic flow”. In: *Journal of Vision* 10.13 (Jan. 2011), pp. 25–25. ISSN: 1534-7362. DOI: `10.1167/10.13.25`.

[108] Rory Sayres et al. “Using a Deep Learning Algorithm and Integrated Gradients Explanation to Assist Grading for Diabetic Retinopathy”. In: *Ophthalmology* 126.4 (2019). ISSN: 15494713. DOI: `10.1016/j.ophtha.2018.11.016`.

[109] R M Seyfarth and D L Cheney. “Production, usage, and comprehension in animal vocalizations”. In: *Brain and Language* 115.1 (2010), pp. 92–100. ISSN: 0093-934X. DOI: `Doi10.1016/J.Bandl.2009.10.003`. URL: `<GotoISI>://WOS:000284438600010`.

[110] R M Seyfarth and D L Cheney. “Vocal Development in Vervet Monkeys”. In: *Animal Behaviour* 34 (1986), pp. 1640–1658. ISSN: 0003-3472.

[111] Robert M Seyfarth et al. “The central importance of information in studies of animal communication”. In: *Animal Behaviour* 80.1 (2010), pp. 3–8. ISSN: 00033472. DOI: `10.1016/j.anbehav.2010.04.012`.

[112] Dinggang Shen, Guorong Wu, and Heung-Il Suk. “Deep Learning in Medical Image Analysis”. In: (2017). DOI: 10.1146/annurev-bioeng-071516. URL: https://doi.org/10.1146/annurev-bioeng-071516-.

[113] Yu Shiu et al. “Deep neural networks for automated detection of marine mammal species”. In: *Scientific Reports* 10.1 (Dec. 2020). ISSN: 20452322. DOI: 10.1038/s41598-020-57549-y.

[114] Karen Simonyan, Andrea Vedaldi, and Andrew Zisserman. “Deep Inside Convolutional Networks: Visualising Image Classification Models and Saliency Maps”. In: (Dec. 2013). URL: http://arxiv.org/abs/1312.6034.

[115] J Soltis. “Vocal Communication in African Elephants (Loxodonta africana)”. In: *Zoo Biology* 29.2 (2010), pp. 192–209. ISSN: 0733-3188.

[116] Jaemin Son et al. “Development and Validation of Deep Learning Models for Screening Multiple Abnormal Findings in Retinal Fundus Images”. In: *Ophthalmology* 127.1 (2020). ISSN: 15494713. DOI: 10.1016/j.ophtha.2019.05.029.

[117] W. W. Sprague et al. “Stereopsis is adaptive for the natural environment”. In: *Science Advances* 1.4 (2015), e1400254–e1400254. ISSN: 2375-2548. DOI: 10.1126/sciadv.1400254. URL: http://advances.sciencemag.org/cgi/doi/10.1126/sciadv.1400254.

[118] Michael W. Spratling. “Predictive coding as a model of response properties in cortical area V1”. In: *Journal of Neuroscience* 30.9 (Mar. 2010), pp. 3531–3543. ISSN: 02706474. DOI: 10.1523/JNEUROSCI.4911-09.2010.

[119] Pei Sun et al. “Scalability in Perception for Autonomous Driving: Waymo Open Dataset”. In: (Dec. 2019). URL: http://arxiv.org/abs/1912.04838.

[120] F E Theunissen, K Sen, and A J Doupe. “Spectral-temporal receptive fields of nonlinear auditory neurons obtained using natural sounds”. In: *J Neurosci* 20.6 (2000), pp. 2315–2331. ISSN: 1529-2401 (Electronic)0270-6474 (Linking). URL: http://www.ncbi.nlm.nih.gov/pubmed/10704507.

[121] Yapeng Tian, Di Hu, and Chenliang Xu. “Cyclic Co-Learning of Sounding Object Visual Grounding and Sound Separation”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition.* 2021, pp. 2745–2754.

[122] Yonglong Tian, Dilip Krishnan, and Phillip Isola. “Contrastive Multiview Coding”. In: (June 2019). URL: http://arxiv.org/abs/1906.05849.

[123] Daniel Shu Wei Ting et al. “Development and validation of a deep learning system for diabetic retinopathy and related eye diseases using retinal images from multiethnic populations with diabetes”. In: *JAMA - Journal of the American Medical Association* 318.22 (2017). ISSN: 15383598. DOI: 10.1001/jama.2017.18152.

[124] Avinash V. Varadarajan et al. "Deep learning for predicting refractive error from retinal fundus images". In: *Investigative Ophthalmology and Visual Science* 59.7 (2018). ISSN: 15525783. DOI: `10.1167/iovs.18-23887`.

[125] Arun Balajee Vasudevan, Dengxin Dai, and Luc Van Gool. "Semantic object prediction and spatial sound super-resolution with binaural sounds". In: *European Conference on Computer Vision*. 2020, pp. 638–655.

[126] A L Vergne, M B Pritz, and N Mathevon. "Acoustic communication in crocodilians: from behaviour to brain". In: *Biological Reviews* 84 (3 2009), pp. 391–411. ISSN: 14647931 1469185X. DOI: `10.1111/j.1469-185X.2009.00079.x`.

[127] Jacob Walker, Abhinav Gupta, and Martial Hebert. *Dense Optical Flow Prediction from a Static Image*. Tech. rep.

[128] Jiayun Wang et al. "A deep learning approach for meibomian gland atrophy evaluation in meibography images". In: *Translational Vision Science and Technology* 8.6 (Nov. 2019). ISSN: 21642591. DOI: `10.1167/tvst.8.6.37`.

[129] Xudong Wang, Ziwei Liu, and Stella X. Yu. *Unsupervised feature learning by cross-level discrimination between instances and groups*. 2020.

[130] Xudong Wang, Ziwei Liu, and Stella X. Yu. *Unsupervised feature learning by cross-level discrimination between instances and groups*. 2020.

[131] W.H. Warren. "Optic Flow". In: *The Senses: A Comprehensive Reference*. Elsevier, 2008, pp. 219–230. DOI: `10.1016/B978-012370880-9.00311-X`.

[132] S M Woolley et al. "Functional groups in the avian auditory system". In: *J Neurosci* 29.9 (2009), pp. 2780–2793. ISSN: 1529-2401 (Electronic).

[133] Zhirong Wu et al. "Unsupervised Feature Learning via Non-parametric Instance Discrimination". In: *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*. 2018. DOI: `10.1109/CVPR.2018.00393`.

[134] Fanyi Xiao et al. "Audiovisual slowfast networks for video recognition". In: *arXiv preprint arXiv:2001.08740* (2020).

[135] Huazhe Xu et al. "End-to-end Learning of Driving Models from Large-scale Video Datasets". In: (Dec. 2016). URL: `http://arxiv.org/abs/1612.01079`.

[136] Qi Yan et al. "Deep-learning-based prediction of late age-related macular degeneration progression". In: *Nature Machine Intelligence* 2.2 (2020). ISSN: 2522-5839. DOI: `10.1038/s42256-020-0154-9`.

[137] Tal Yarkoni and Jacob Westfall. "Choosing Prediction Over Explanation in Psychology: Lessons From Machine Learning". In: *Perspectives on Psychological Science* 12 (6 Nov. 2017), pp. 1100–1122. ISSN: 1745-6916. DOI: `10.1177/1745691617693393`.

[138] Malcolm P Young and Shigeru Yamane. "Sparse population coding of faces in the inferotemporal cortex". In: *Science* 256.5061 (1992), pp. 1327–1331.

[139] Fisher Yu et al. “BDD100K: A Diverse Driving Dataset for Heterogeneous Multitask Learning”. In: (May 2018). URL: `http://arxiv.org/abs/1805.04687`.

[140] Huai Yu et al. “A color-texture-structure descriptor for high-resolution satellite image classification”. In: *Remote Sensing* 8.3 (2016). ISSN: 20724292. DOI: `10.3390/rs8030259`.

[141] Matthew D Zeiler and Rob Fergus. “Visualizing and Understanding Convolutional Networks”. In: (Nov. 2013). URL: `http://arxiv.org/abs/1311.2901`.

[142] Richard Zhang, Phillip Isola, and Alexei A. Efros. “Colorful image colorization”. In: *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*. Vol. 9907 LNCS. 2016. DOI: `10.1007/978-3-319-46487-9{\_}40`.

[143] Zhoutong Zhang et al. “Shape and material from sound”. In: (2017).

[144] Liang Zheng, Yi Yang, and Qi Tian. “SIFT Meets CNN: A Decade Survey of Instance Retrieval”. In: (Aug. 2016). URL: `http://arxiv.org/abs/1608.01807`.

[145] Ming Zhong et al. “Beluga whale acoustic signal classification using deep learning neural network models”. In: *The Journal of the Acoustical Society of America* 147.3 (Mar. 2020), pp. 1834–1841. ISSN: 0001-4966. DOI: `10.1121/10.0000921`.

[146] Ming Zhong et al. “Detecting, classifying, and counting blue whale calls with Siamese neural networks”. In: *The Journal of the Acoustical Society of America* 149.5 (May 2021), pp. 3086–3094. ISSN: 0001-4966. DOI: `10.1121/10.0004828`.

[147] Ming Zhong et al. “Multispecies bioacoustic classification using transfer learning of deep convolutional neural networks with pseudo-labeling”. In: *Applied Acoustics* 166 (Sept. 2020), p. 107375. ISSN: 0003682X. DOI: `10.1016/j.apacoust.2020.107375`.

[148] Bolei Zhou et al. *Learning Deep Features for Discriminative Localization.* Tech. rep. URL: `http://cnnlocalization.csail.mit.edu`.

[149] Bolei Zhou et al. “Object detectors emerge in deep scene CNNs”. In: *3rd International Conference on Learning Representations, ICLR 2015 - Conference Track Proceedings.* 2015.

[150] Bolei Zhou et al. *Temporal Relational Reasoning in Videos.* Tech. rep. URL: `http://relation.csail.mit.edu/.`.