UNIVERSITY OF CALIFORNIA SAN DIEGO

Investigating the effectiveness of Log-Polar projections in conjunction with Convolutional Networks

A thesis submitted in partial satisfaction of the
requirements for the degree Master of Science

in

Computer Science

by

Mrigankshi Kapoor

Committee in charge:

Professor Garrison Cottrell, Chair
Professor Taylor Berg-Kirkpatrick
Professor Virginia de Sa

2024

The Thesis of Mrigankshi Kapoor is approved, and it is acceptable in quality and form for publication on microfilm and electronically.

University of California San Diego

2024

TABLE OF CONTENTS

LIST OF FIGURES

ACKNOWLEDGEMENTS

I would like to acknowledge Professor Garrison Cottrell for his support as the chair of my committee. Through multiple drafts and many long nights, his guidance has proven to beinvaluable.

I would also like to acknowledge Professor Virginia de Sa and Professor Taylor Berg-Kirkpatrick. Their support as thesis committee members and their feedback on this work have helped me improve my writing.

A sincere thank you to the GURU lab, and especially past members Shubham Kulkarni and Martha Gahl, without whom my research would have no doubt taken fives times as long. Their support has motivated me to never stop trying to improve, no matter how many failures I encountered. Their past work set the direction for my own research, and their insights have significantly shaped my approach.

Finally, thank you to every person who has supported and encouraged me to pursue this thesis. To my parents: Thank you for supporting me through all the highs and lows. I am ever so grateful. To my friends: I am so fortunate to have met such supportive and kind people during my time here at UC San Diego.

ABSTRACT OF THE THESIS

Investigating the effectiveness of Log-Polar projections in conjunction with Convolutional Networks

by

Mrigankshi Kapoor

Master of Science in Computer Science

University of California San Diego, 2024

Professor Garrison Cottrell, Chair

This study explores biologically inspired transformations and training techniques for image networks, such as log-polar projections and curriculum learning, in conjunction with convolutional neural networks (CNNs). Specifically, it presents a comparative analysis of log-polar CNNs and traditional CNN architectures. The key difference in log-polar CNNs is the conversion of input images from Cartesian to polar coordinates, followed by a logarithmic transformation of the radial coordinate *'r'*. Preliminary experiments indicate that log-polar CNNs exhibit enhanced robustness to rotation and scale changes at inference time when trained with log-polar transformed images. Additionally, our results highlight improved resilience to

geometric distortions and specific noise types, suggesting potential for broader applications in adversarial and biologically inspired modeling tasks. This research also investigates ways to align these log-polar networks and their representations with the human visual system's learning mechanisms, aiming to achieve superior overall performance under standard conditions.

Introduction

Primate visual learning is highly efficient, outperforming traditional deep convolutional neural networks (CNNs) in several key areas. One major advantage of the primate visual system is its ability to handle variations in visual inputs, such as rotation and scaling, without the need for extensive training. In contrast, CNNs often require large amounts of data augmentation to achieve invariance to these transformations. This augmentation process is time-consuming and does not fully replicate the natural robustness of biological systems.

Similarly, CNNs are vulnerable to small, human-imperceptible perturbations in input data, as highlighted in [21]. To make CNNs more robust to noise, they must be explicitly trained on adversarial examples, a process that adds complexity and does not guarantee reliability when faced with new or unforeseen perturbations, as demonstrated in the seminal work on adversarial training by Goodfellow et al. [10].

Additionally, there are common neurological phenomena that CNNs fail to explain. This indicates that CNNs do not fully replicate how the human visual system processes complex visual information, limiting their effectiveness in tasks that require human-like perception. One such phenomenon is the face inversion effect, which demonstrates how humans struggle to recognize faces when they are inverted [24]. Unlike humans, current CNN architectures cannot fully account for this effect, as shown in recent studies including the work by Baker and Elder [2].

Inspired by these observations, we aim to design networks that more closely mimic the human visual system by exploring neurobiologically inspired computation. Two such mechanisms—log-polar transformations and curriculum learning—offer promising solutions

to the challenges mentioned above. The log-polar transform provides natural rotation and scale invariance, while also allowing networks to focus on different regions of an image based on fixation points, similar to the human visual system. Curriculum learning introduces tasks of increasing difficulty, much like how humans learn, improving generalization and training efficiency [3].

These methods, with their unique properties, have the potential to enhance the performance and robustness of CNNs, enabling them to better handle image transformations and adversarial noise. A more detailed discussion of these approaches is provided in the next chapter.

Chapter 1

Background

1.1 Log Polar Transformation

The log polar transform plays a fundamental role in the human visual system, approximating the mapping from the visual field to the primary visual cortex as explained in Araujo et al. [1] and Polimeni et al. [17].

The log-polar transform converts Cartesian coordinates (x, y) to log-polar coordinates (ρ, θ). The equations are given below and illustrated in Figure 1.1 taken from Sarvaiya et al. [19].

$$\rho = \log\left(\sqrt{x^2 + y^2}\right) \tag{1.1}$$

$$\theta = \arctan\left(\frac{y}{x}\right) \tag{1.2}$$

The reason we are interested in studying log polar transformed input in conjunction with conventional CNN architectures of vision is because of some interesting properties of this transform.

- Image rotation corresponds to a small vertical translation in log-polar space and image scaling corresponds to a horizontal translation, as illustrated in Figure 1.2. We know that convolutional networks lack any rotation or scale invariance but exhibit some degree of

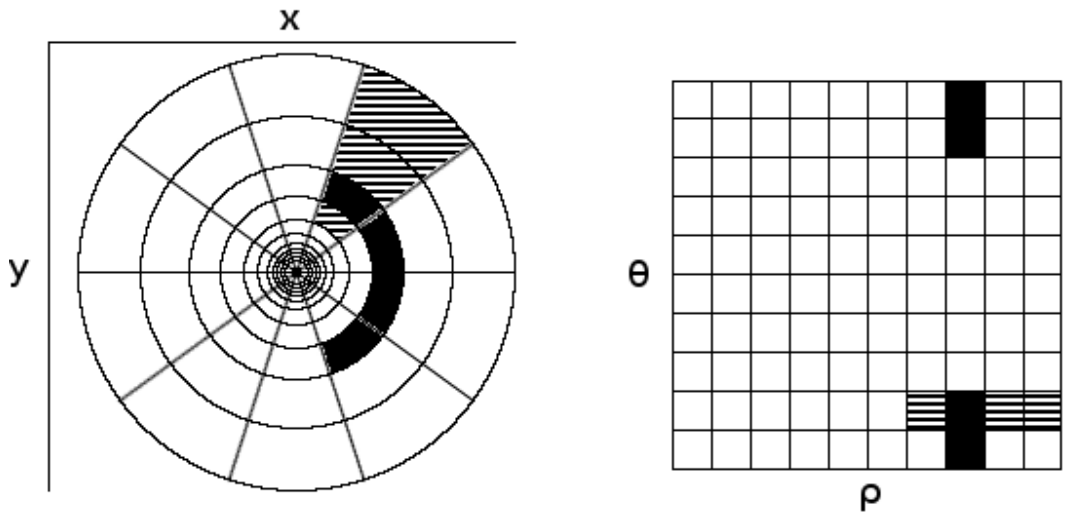

Figure 1.1. Mapping from Cartesian (left) to log polar coordinates (right)

translation invariance. The Log polar transform, coupled with traditional convolutional networks, introduces rotation and scale equivariance to the network. One important thing to note is that, as a trade-off, translation invariance in the original domain is lost when applying the log-polar transform.

- About half of the log-polar space would be filled by the central 2% of a Euclidean image as shown in Figure 1.3. Therefore, this transformation can change the information density in an image based on the where the center for the transform is fixated. This characteristic has great potential to improve the training of the network. By exposing it to different fixations, we can make it focus on different areas of the image and learn to generalize patterns irrespective of the specific location within an image.

Previous work has demonstrated the adequacy of the log-polar transformation as an approximation of how the visual field maps onto the primary visual cortex in primates, as well as in computational models for facial recognition. Gahl et al. [8] offers a comprehensive comparison of CNN performance on Euclidean versus log-polar transformed images when evaluated on inverted images. The first experiment in this study investigates two datasets: one is used to train

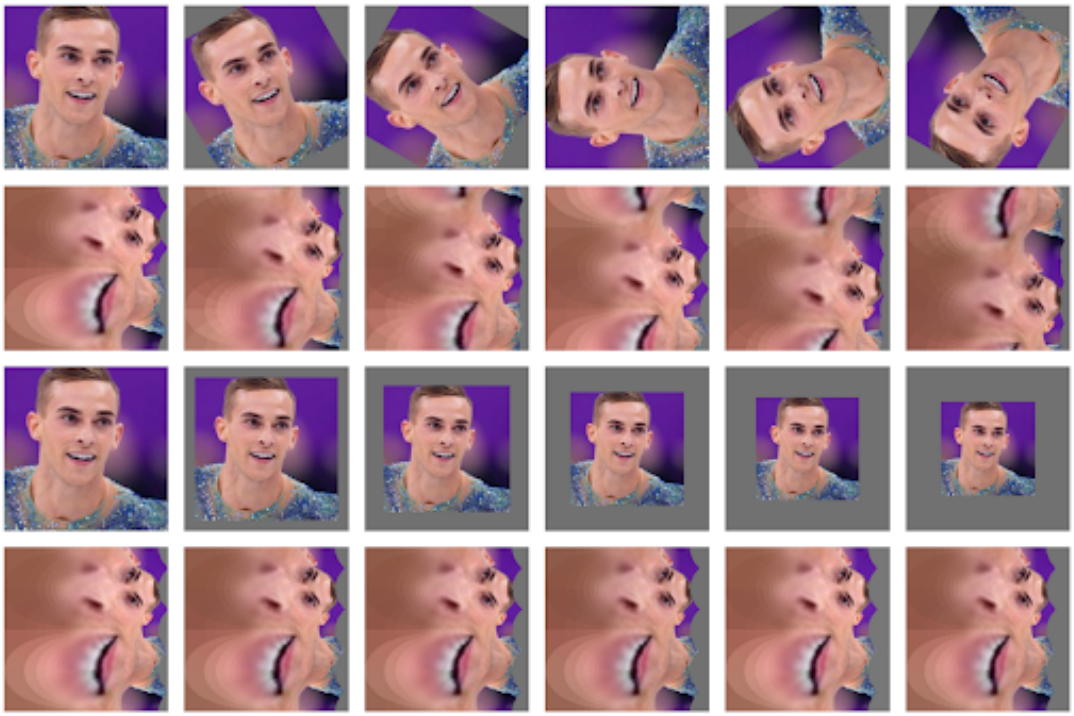

Figure 1.2. Rotation and Scale in Log Polar Space. Top two rows correspond to image rotations in Euclidean and Log Polar space. Bottom two rows correspond to image scale changes in Euclidean and Log Polar space.

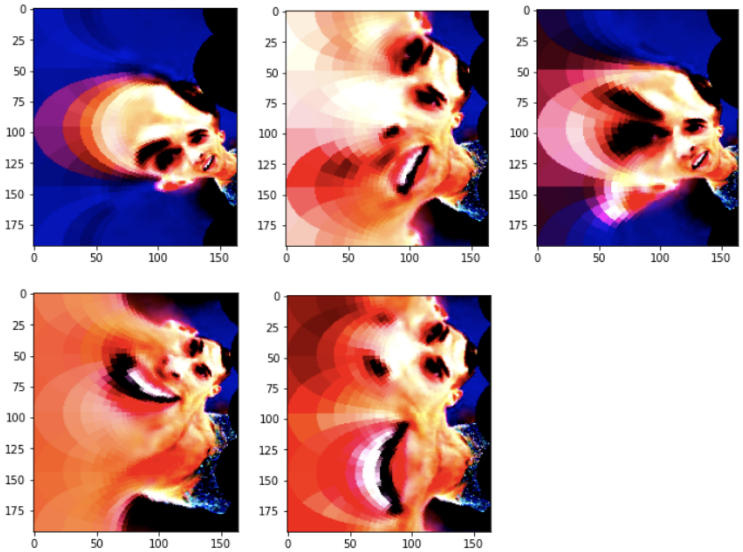

Figure 1.3. Log Polar transform with different fixation points

a model for expertise in facial recognition, while the other is used to train a separate model to achieve a basic-level understanding of mono-oriented objects, such as cars, which are typically viewed from a single orientation. The second experiment compares two models both trained in expertise on cars and dogs. The overall findings reveal that while log-polar models perform slightly worse or similarly to Euclidean models on upright validation images, they achieve significantly higher accuracy when classifying inverted images.

Remmelzwaal et al. [18] evaluates networks trained with the log-polar method for MNIST classification. They present the performance gap between log-polar and Euclidean networks across different image scaling and rotation scenarios. Employing a shallow six-layered network for this task, the log-polar model generally outperforms traditional CNNs, exhibiting an accuracy increase of up to 28%.

In a similar vein, Cao et al. [4] use the log-polar transform for image segmentation tasks using the PASCAL VOC 2007 and PASCAL VOC 2012 datasets. Their approach involves transforming feature maps from a convolutional network into log-polar space instead of the input images themselves, followed by segmentation using a single-shot detector model with a 16-layer VGG backbone. The segmented results are then transformed back into Euclidean space. Additionally, they investigate the effectiveness of employing multiple log-polar projections of the same image, each centered at different fixation points. This research demonstrates improvements over the state-of-the-art baseline for segmentation R-CNN in both upright and rotated evaluation scenarios.

Su and Wen [20] present an innovative approach in which they create a convolutional kernel specifically designed for log-polar space, rather than opting to convert the original image into log-polar coordinates. Unlike standard convolution, which operates on rectangular portions of an image by element-wise multiplication with a parameter matrix W followed by summation, their log-polar space convolution (LPSC) method extracts elliptical portions of the image, projects them onto log-polar space, and then applies the standard convolution operation. Importantly, while the parameter matrix is trained in log-polar space, the image itself is kept in Euclidean

space. This LPSC technique is integrated into various models, including AlexNet, VGG, and ResNet, and evaluated across multiple datasets such as CIFAR-10, CIFAR-100, DRIVE, and ImageNet. Across all datasets, LPSC-enhanced models demonstrate superior performance, although in some cases, the improvements are only marginal.

Previous works have shown the potential of log-polar transformations in specific tasks, such as improving robustness to rotation, scale, and inversion, or for specialized applications like segmentation. In this thesis, we explore whether combining log-polar transformations with other biologically inspired methods can enhance their effectiveness in standard settings and for a broader range of distortions, making them more generally applicable in improving the performance and robustness of CNNs.

1.2 Curriculum Learning

Curriculum learning has emerged as a powerful paradigm in machine learning, drawing inspiration from cognitive science and human learning processes. Its premise lies in the notion that learning a task is more efficient when the training data is presented to the model in a meaningful order, gradually increasing in complexity or relevance. This approach mirrors how humans naturally learn, where concepts are introduced incrementally, building upon foundational knowledge.

McClelland and Rogers [14] conducted seminal research demonstrating that human learning of classification problems follows a coarse-to-fine structure. This finding suggests that humans tend to grasp broad concepts before delving into finer details. Furthermore, semantic dementia was shown to follow a reverse order of degradation.

In a similar vein, Elman [7] highlights the importance of starting with simpler tasks for networks. By starting with easier, foundational tasks, networks can build essential knowledge, which enables them to more effectively progress to more complex problems.

Years later, Bengio et al.[3] formalized this principle into the curriculum learning frame-

work. Proposed as a method to improve neural network training, it was inspired by observations of human learning. By presenting training examples in a meaningful order, starting with easier instances and gradually introducing more challenging ones, curriculum learning aims to guide the model towards better generalization and faster convergence. There can be several ways of ordering the information learned for a classification task, but some common ways are to start with coarser classes or fewer identities.

In this thesis, we apply curriculum learning to enhance the training of our log-polar networks. We begin with a smaller set of identities and gradually increase the number of identities as training progresses. This progressive complexity allows the network to learn more intricate features over time. The idea is that since the log-polar transform is biologically inspired, it should pair well with a training schedule that mimics human learning. We will talk about the details of this approach in the section Experiments.

Chapter 2

Method

2.1 Data

We primarily use the faces dataset for all of our experiments. The dataset was gathered by the GURU Lab at UCSD for use in Gahl et al. [9]. It consists of 150-200 close up images for 128 distinct people. Moreover, we use the default train-validation-test split which contains 17260 train images, 2159 validation images, and 2160 test images. Example images from this dataset can be found in Figure 2.1. The faces dataset is a relatively small dataset, with only 170 images per identity on average. The results from our analysis can later be verified with larger and more diverse datasets, such as Imagenet subsets.

2.2 Transformations

We have experimented with several transformations similar to the SimCLR pipeline from Chen et al. [5] which performs random crops, horizontal flip, color jitter, color grayscale, and random Gaussian blurring. We modify this pipeline by introducing a way to foveate images and move them to log polar space in addition to the transformations in the SimCLR pipeline.

We primarily rely on the standard PyTorch Torchvision library transforms for our image transformations. However, as the library lacks implementations for polar projection and foveation, we encounter a limitation. To address this, we incorporate the foveation technique described by Perry and Geisler [16]. Additionally, we take on the task of implementing polar and log-polar

Figure 2.1. Examples from Faces Dataset

projections based on methodologies outlined in Thunuguntla [22], Zheng and Matungka [**?**], and van der Walt et al. [23]. Utilizing CUDA-compatible PyTorch code, we ensure that these transformations are tailored for GPU acceleration, optimizing computational performance.

The log-polar and polar transformations, being non linear mappings, introduce distortions when mapping Cartesian space, resulting in various artifacts in the transformed images. To mitigate these issues and enhance image quality, we employ diverse strategies. Firstly, we implement a masking technique to exclude pixels in the transformed image corresponding to areas beyond the original image boundaries. Secondly, we explore smoothing methods such as nearest-neighbor and distance-based approaches, akin to bilinear smoothing, to further refine the transformed images. Additionally, we experiment with different projection techniques, including circumscribed and inscribed projections, to optimize the final output. Given the circular nature of polar or log-polar projections, we have the flexibility to choose between inscribing or circumscribing the circular projection within the preimage. What that means is, in the case of inscribed projection, the circular log-polar region is completely contained within the input image with Inscribed projection. So the radius of the circle is limited to fit entirely inside the rectangular bounds of the input image. In the case of circumscribed projection, The circular

log-polar region extends to cover the entire image, including the corners. The radius of the circle is large enough to encompass the entire rectangular input and can go out of its bounds. Figure 2.2 illustrates these different strategies to remove transformation artifacts. Given input shape (H,W) and output shape (M,N), the log-polar transformation and mapping can be described as follows:

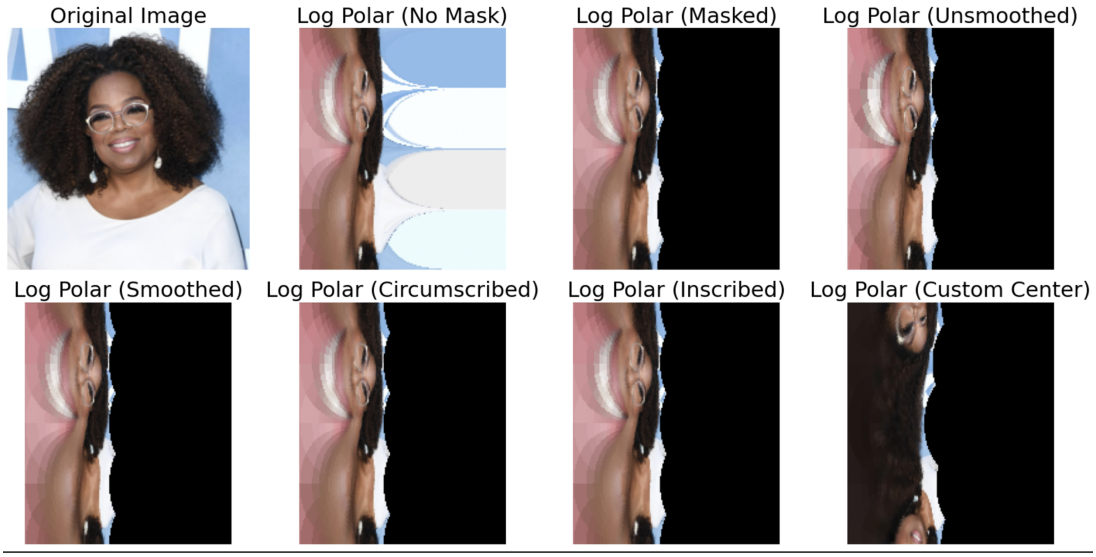

Figure 2.2. Visualizing Log-Polar Transformations: Impact of Masking, Smoothing, and Projection Techniques

1. Compute the maximum radius $R_{\max}$ based on whether the position is 'circumscribed' or not:

$$R_{\max} = \log\left(\frac{\|[H,W]\|_2}{2} \times \text{log\_polar\_distance}\right) \tag{2.1}$$

$$\text{or} \quad R_{\max} = \log\left(\frac{\max(H,W)}{2} \times \text{log\_polar\_distance}\right) \tag{2.2}$$

2. Create a meshgrid for the output shape (M,N):

$$\theta_{ij} = \frac{2\pi i}{M}, \quad i = 0,1,\ldots,M-1 \tag{2.3}$$

$$r_{ij} = \frac{jR_{\max}}{N}, \quad j = 0,1,\ldots,N-1 \tag{2.4}$$

3. Mapping Log-Polar mesh to Cartesian coordinates:

$$x_{ij} = \exp(r_{ij})\cos(\theta_{ij}) \tag{2.5}$$

$$y_{ij} = \exp(r_{ij})\sin(\theta_{ij}) \tag{2.6}$$

4. Centering: If (c_x, c_y) is the center of the transformation (which can be a given point or chosen probabistically), center the map such that (c_x, c_y) is the origin of the log polar space:

$$X = c_x + x_{ij} \tag{2.7}$$

$$Y = c_y - y_{ij} \tag{2.8}$$

The reason we allow for choosing a center probabilistically is because we also use log polar transform around a saliency-based fixation. What this means is that we pick a point from the saliency map as the fixation center, treating the saliency map as a probability distribution. We use opencv's implementation of the method described in Hou et al. [12] to create a saliency map for all the input images which is used as a probability distribution to select the fixation point. We also compared this method with other modern Saliency map calculations like DeepGaze from [15] but Hou's method gives better results for faces, where it highlights prominent features such as eyes,nose, mouth's in the salience map as illustrated in Figure 2.3

2.3 Training

NRP Nautilus furnishes the computational resources for our tasks. Standardized training occurs on systems equipped with 8 CPU cores, 24 GB of RAM, and a single NVIDIA A10 GPU. Training is conducted using mixed precision (FP16) in PyTorch Lightning, which helps to optimize computational resources. Our setup utilizes PyTorch version 1.10.0 with CUDA version

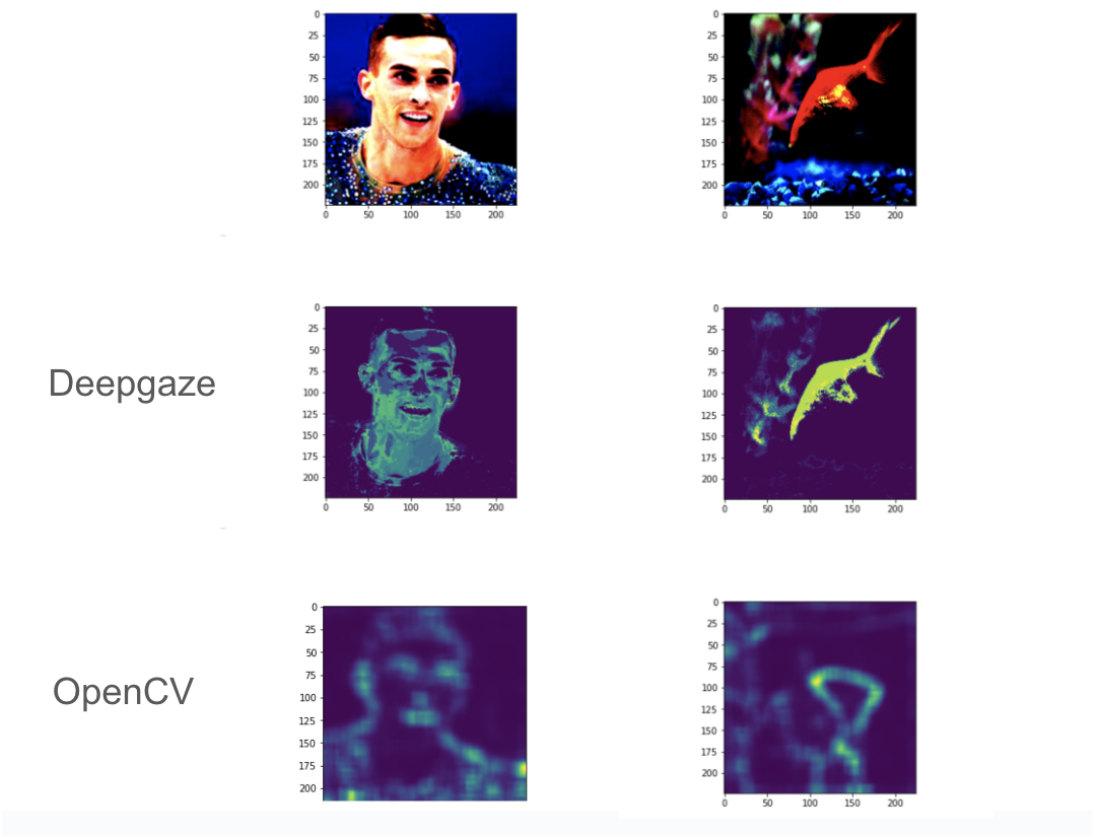

Figure 2.3. Comparison of DeepGaze and OpenCV Salience Functions on Faces and Objects

11.3 and CUDA compute capability 8.0. Tensor cores are leveraged to expedite the training process. Training is distributed across 8-node clusters, with distributed sampling managed by PyTorch DDP and its high level APIs furnished in PyTorch Lightning.

2.4 Evaluation

We use a similar approach as in Remmelzwaal et al. [18] to compare our Log polar trained model with regular CNNs. Once we have trained the models as desired, we then evaluate the models against 25 different rotated and scaled variants of each validation and testing image. Rotation varies between 0 and 180 degrees (full inversion), and scale values between 0.5 (half-size) and 1.0 (original size).

Chapter 3

Experiments

3.1 Faces Dataset with ResNet and VGG

We first verify the results from Remmelzwaal et al. in [18] on our faces dataset employing deeper networks and convolutional networks with residual connections. We choose ResNet18 and VGG16 for their simple architecture and popularity in classification applications.

We train both of these models with and without augmentation and evaluate them as explained in the section above to construct the accuracy heatmaps. For the model trained on augmented data, we use conventional CNN augmentation techniques such as horizontal flipping, color jittering and small random rotation on every image to train the Euclidean model. In contrast, for the log polar counterpart, the input undergoes augmentation through a saliency-based fixation selection approach described in the Transformation section. We create a saliency map for all input images, treating it as a probability distribution. A point is chosen from this distribution, with the sampling probability equal to the distribution's probability density. This selected point is used as the fixation for the log polar transform. We can thus generate multiple augmented images from one. Since we are dealing with faces, this methodology is anticipated to produce transformations centered around crucial facial features such as eyes and nose and result in augmented images similar to the ones in Figure 1.3.

Figure 3.1 and 3.2 show the accuracy heatmaps to compare the performance of Euclidean and Log Polar networks. Each cell within the heatmap corresponds to a specific rotation angle

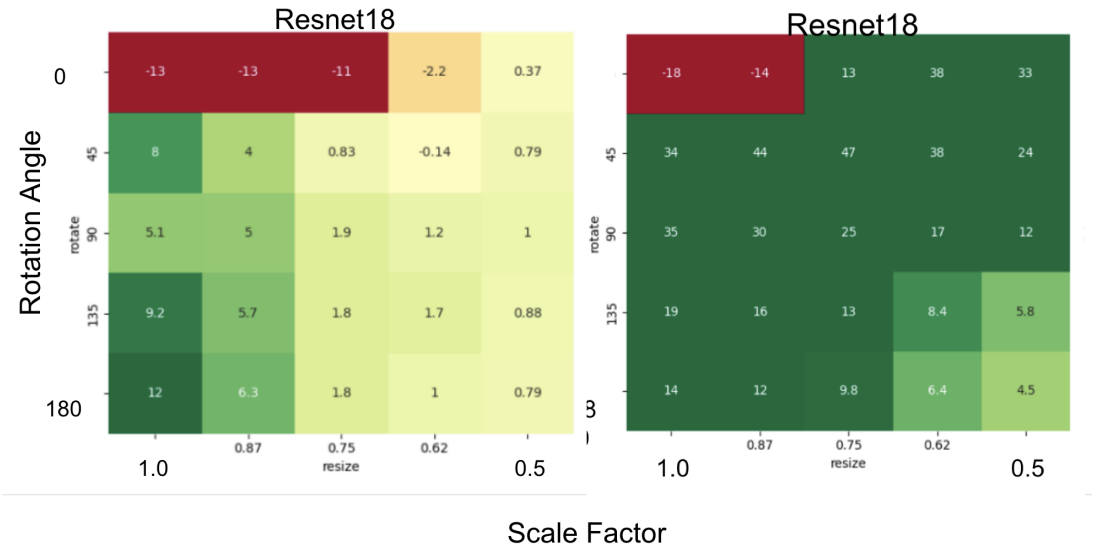

Figure 3.1. Accuracy Heatmap for Resnet18 trained with (right) and without (left) augmentation

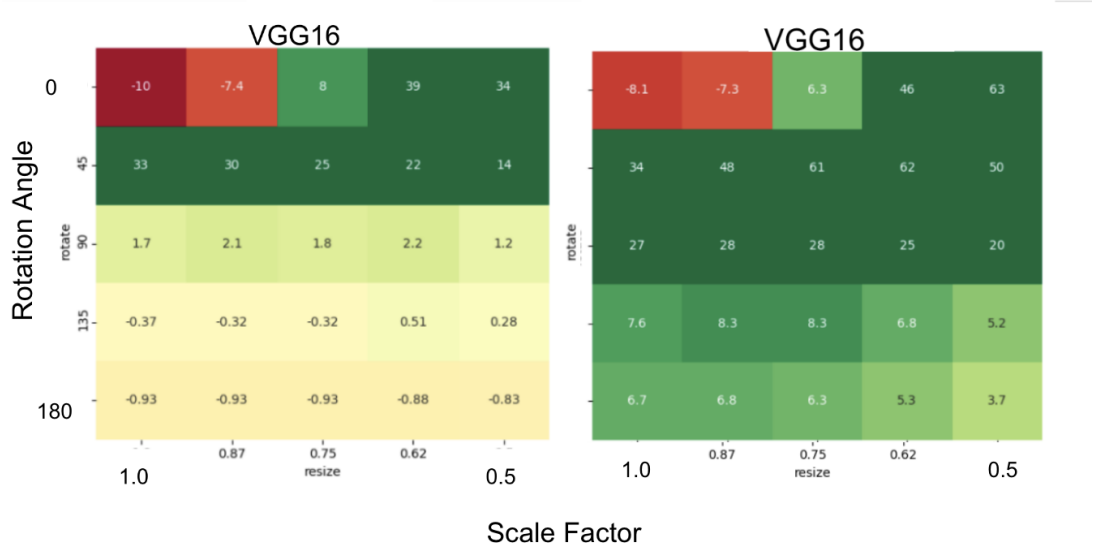

Figure 3.2. Accuracy Heatmap for VVG16 trained with (right) and without (left) augmentation

and a scale factor by which the validation dataset is transformed before evaluation. The value at the cell represents the difference in accuracy between log polar and Euclidean models. A green cell indicates that the log polar model outperforms the Euclidean network, while a red block indicates otherwise.

Our analysis of the heatmap reveals a discernible pattern: the Log Polar Network shows high robustness to moderate scale/rotation changes at evaluation time. We are able to further

enhance accuracy gains through image augmentation techniques. However, all the CNN-based architectures we trained with Euclidean and Log polar inputs, struggled to outperform Euclidean models under low or no variation in scale/rotation.

In the following study, we explore different approaches to better align our networks and representations with the learning mechanisms found in the human visual system. Our aim is to achieve a green cell at the top left corner of our accuracy heatmaps, in other words we want to achieve better performance by the log polar networks over Euclidean models in the standard setting.

3.2 Multitask Network

As detailed in Colby's work [6] and similar studies, the parietal cortex has multiple representations of visual input connected to different tasks/actions that we can perform. Among these, salience-based representations are prominent and play a crucial role in determining subsequent eye movements and are also associated with spatial memory.

Based on this understanding, we hypothesize that by incorporating an image regeneration loss into our classification network, which generates a log polar image fixated at a given coordinate and given the original image, and supervising the network on a combined loss we can obtain representations that more closely resemble those found in primate visual systems.

We compare two networks as done previously - one trained on Euclidean images and the other one on log polar images. The Euclidean network is fed the Euclidean image input (X_1) and a random x-y coordinate (p) from within its pixels along with the target label (Y_{label}) and target image for regeneration (Y_{image}), which is the log polar transformation of the original image X_1 fixated at point p. Figure 3.3 provides a visual representation of this network architecture.

Similarly, the log polar network takes a log polar transformed image of the original dataset (fixated at the center of the image) (X_2). p, Y_{image}, and Y_{label} remain the same as before. Note that Y_{image} will be the log polar transformation, fixated at p, not of X_2 directly, but rather of

the corresponding Euclidean image. Figure 3.4 illustrates this setup.

Both networks share the same encoder-decoder architecture. This consists of a VGG16 or Resnet18 encoder followed by a series of alternating transposed convolution and Batch Norm layers for the decoder. The classification features obtained at the end of the encoder are utilized for calculating the classification loss. These features are concatenated with the point (p) before being passed into the decoder for regeneration. The regeneration loss is determined by the pixelwise mean squared error (MSE) loss between the target image and the regenerated image.

To combine these losses, we employ the following equation:

$$Loss = \lambda L_{CE} + (1 - \lambda) L_{MSE} \tag{3.1}$$

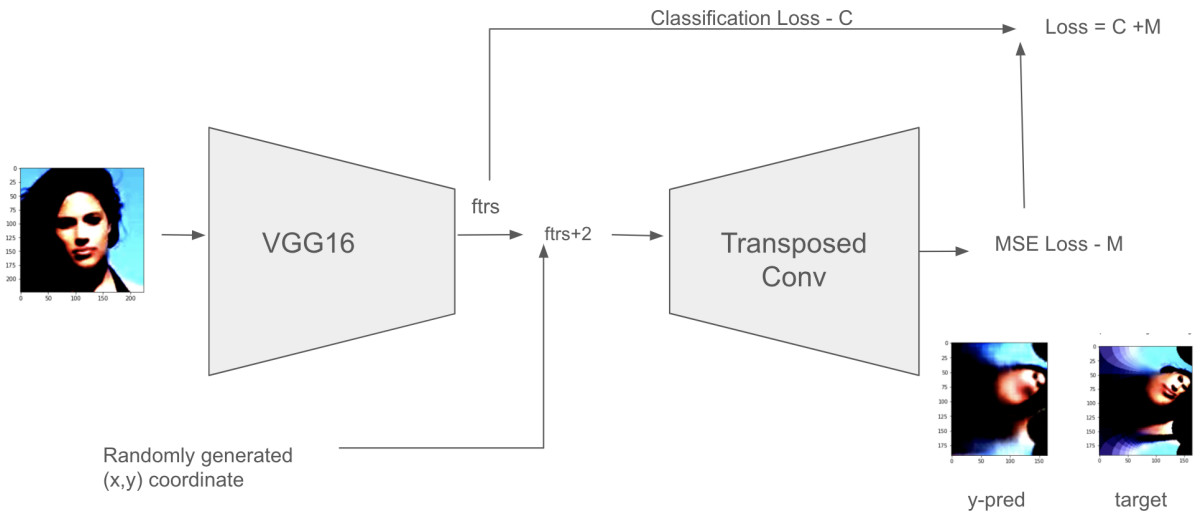

Figure 3.3. Multitask Network training with Euclidean images

Figure 3.5 illustrates the classification training loss and validation accuracy for these Multitask networks. It is evident that while the validation accuracy for the log polar network improves with the multitask approach, the Euclidean network still maintains superior performance. Despite experimenting with various network adjustments such as different hyperparameters and batch normalization layers, the overall trend persists.

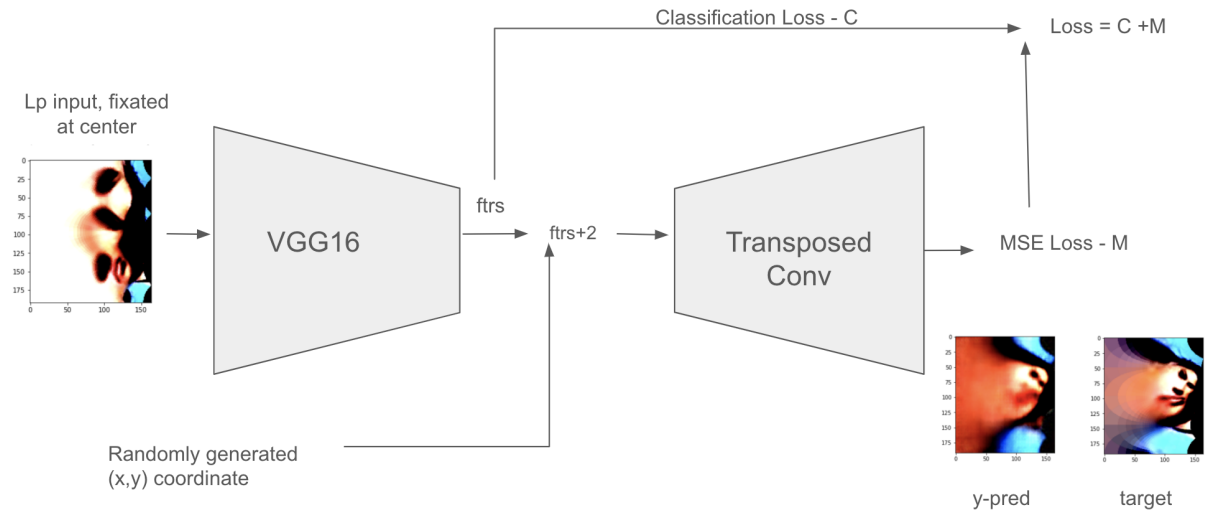

Figure 3.4. Multitask Network training with Log Polar images

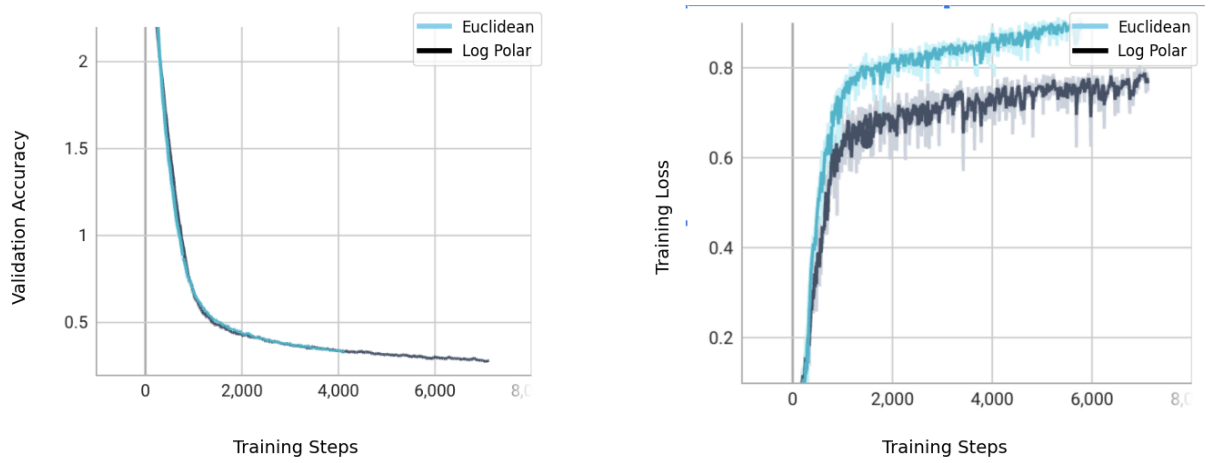

Figure 3.5. Validation Accuracy Plots - Multitask network

3.3 Emulating distance in images

In our previous experiments, we have been using the images from the faces dataset without extensive scaling. These images simulate viewing a face at close range, where the face occupies a significant portion of the visual field. However, in real-life scenarios, faces are often observed from a distance, where, on average, they subtend a visual angle of around 8 degrees in conversational settings. Consequently, the central region of the image contains the majority of facial features and there should be little variation in log polar transformed images fixated at

different features of the face.

To address this, we explore various methods. The simplest approach involves scaling the image to 1/4th or 1/8th of its original size and padding the remaining dimensions with white pixels as shown in going from the top row to the bottom row in Figure 3.6. However, this approach leads to a notable loss in resolution, limiting the information available for CNN learning, especially in higher-frequency details of the face. Another option involves adjusting the radial axes range in the log-polar coordinates of the image to fit most of the central image within the left half, as depicted in Figure 3.6.

Another option involves adjusting the radial axes range in the log-polar coordinates of the image to fit most of the central image within the left half, as depicted in going from left to right in Figure 3.6. In this transformation, the radial axes (representing the distance of a pixel from the center of the image) are scaled such that central area of the image that contains the important face features occupies a smaller region of the visual field.

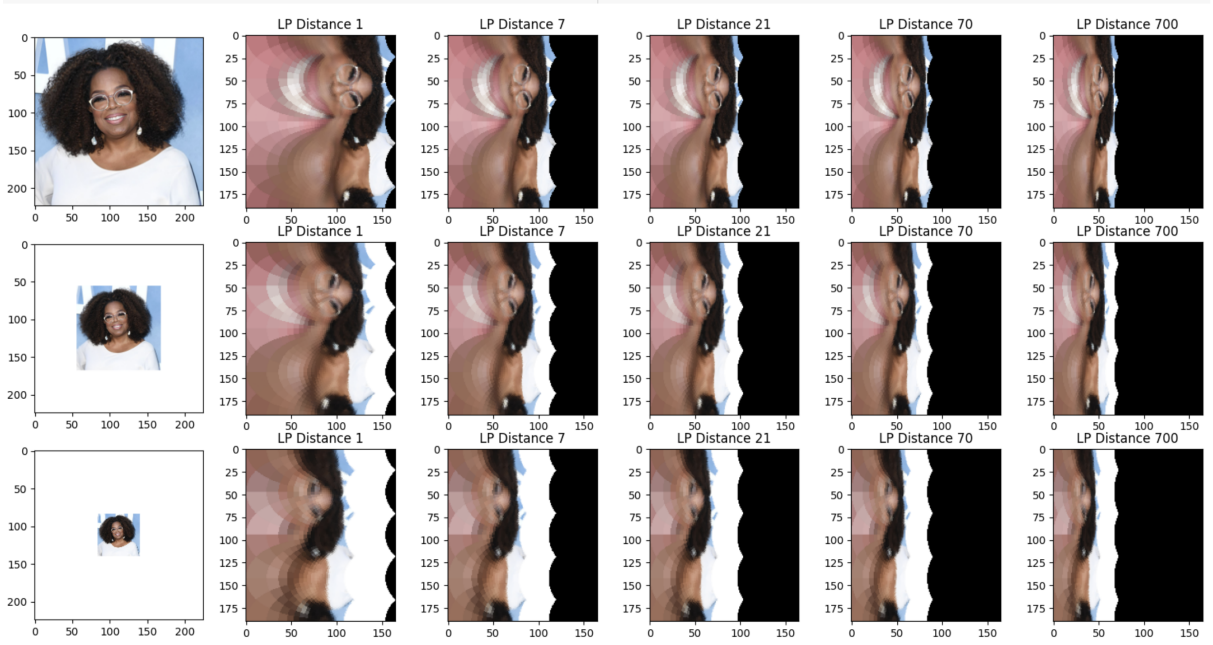

Figure 3.6. Log Polar transform on different scales (y axis) and different ranges of radial distance (x axis)

We run experiments to compare the difference in performance of a resnet model trained

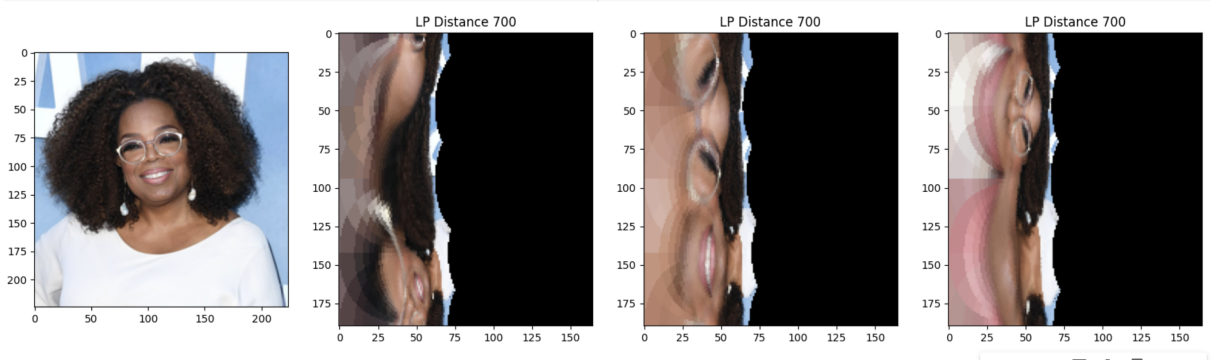

Figure 3.7. Log Polar transform with different fixation points using a large range of radial distance

with euclidean images and log polar images that are transformed this way and the results are presented below in 3.8. The model trained on log polar images fails to outperform its Euclidean counterpart, as evidenced by the validation accuracy plots - the blue plots corresponding to Euclidean networks.

This performance gap could be due to several factors. First, despite the central focus on the face in the log-polar transformation, we still see a fair amount of variation in the vertical axis (the theta axis) of the log-polar images when we use different fixation points as in Figure 3.7. Moreover, because of the circular nature of the transformation, scaling the radial axes leads to a lot of pixels that now have no useful information for classification.

Given these findings, we conclude that the log-polar transformation may not be fully suitable for our current dataset. It is an interesting approach to model realistic viewpoints, but the current limitations in resolution may hinder its performance. Thus, we will need to revisit this analysis when we have access to a high-resolution dataset that better represents faces observed at varying distances and with more refined details.

3.4 Curriculum Learning with Blurring

To mimic the incremental learning process observed in humans, we employ the curriculum learning framework. We begin with four identities, mirroring the early stages of human

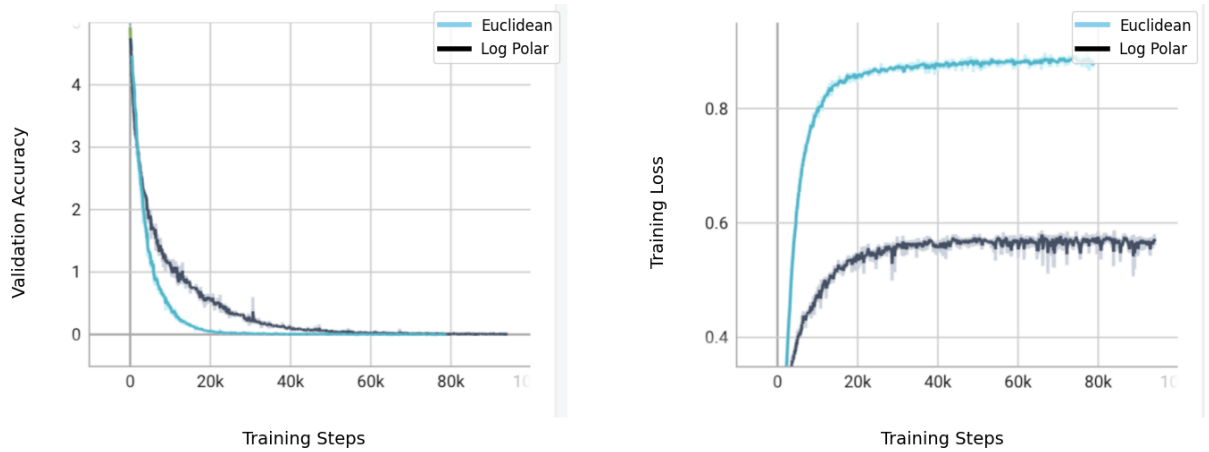

Figure 3.8. Validation Accuracy Plots - radial distance experiments

face recognition, where infants typically recognize familiar faces before expanding to a broader range. We then incrementally increase the number of identities to 8, 16, 32, 64, and eventually 128, simulating a progressive learning process. Just as infants initially perceive the world in low resolution, we start with a Gaussian blur with a sigma of 5 and progressively decrease it to 0, transitioning through values like 2.5 and 1.25, similar to the approach described in Jinsi et al. [13].

Again, we train our model with and without augmentation. We train one model with no augmentation of the dataset and one with random crops for the Euclidean model and fixation based augmentation for Log Polar network. The validation accuracy plots for these trained models are depicted in Figure 3.9,

These trends suggest that the gap in validation accuracy between the two models narrows within the curriculum learning framework compared to our previous experiments. Validation accuracies are generally higher with augmentation for both models. Interestingly, the Log Polar model is able to match the accuracy of the Euclidean model in the early stages of training (with 4/8 identities). However, as the training progresses and the number of identities increases, the gap widens, with the Euclidean model outperforming the Log Polar model.

This suggests that the Log Polar model is effective in simpler scenarios with fewer

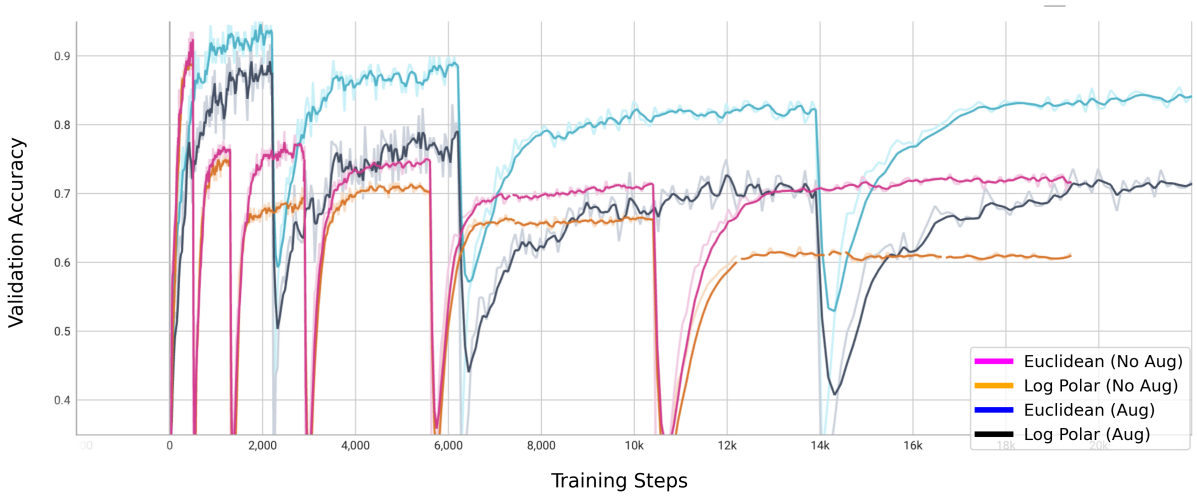

Figure 3.9. Validation Accuracy Plots - Curriculum Learning

identities, where it likely focuses on more coarse, high-level features that are sufficient for accurate classification. However, as the number of identities increases, the model needs to learn finer-grained features that are important for handling a larger and more diverse identity set. In this scenario, the Log Polar model struggles to adapt and capture these detailed variations. While further exploration of design choices could help enhance the Log Polar model's ability to capture finer features, it is possible that some of this loss of high frequency details is inherent to the nature of the transformation.

Chapter 4

Evaluation

In this chapter, we evaluate the experiments conducted in the previous section. In addition to verifying Remmelzwaal's results on rotational and scaled inference across our dataset and various CNN architectures, we explored three key experiments in the previous chapter: Multitask Learning, Radial Distance Emulation, and Curriculum Learning.

Out of the three experiments, we decided to defer Radial Distance Emulation analysis for later as it was not suitable with the current setup. For the remaining two—Multitask Learning and Curriculum Learning—we did not observe consistently outperforming validation accuracy for the log polar models compared to the Euclidean models in the standard evaluation setting. However, in this section, we focus on analyzing the accuracy trends of these models when subjected to rotated, scaled, and corrupted inputs.

4.1 Accuracy Heatmaps

In this section, we present the accuracy heatmaps for the networks in consideration with different scales and rotations of test images, just like the accuracy maps we saw earlier.

Figures 4.1 and 4.2 illustrate the accuracy heatmap of the ResNet model trained using the curriculum learning schedule described above. The conditions for each model are as follows:

1. With only small random rotation augmentation (-15 to 15 degrees) applied to both Log Polar and Euclidean models.

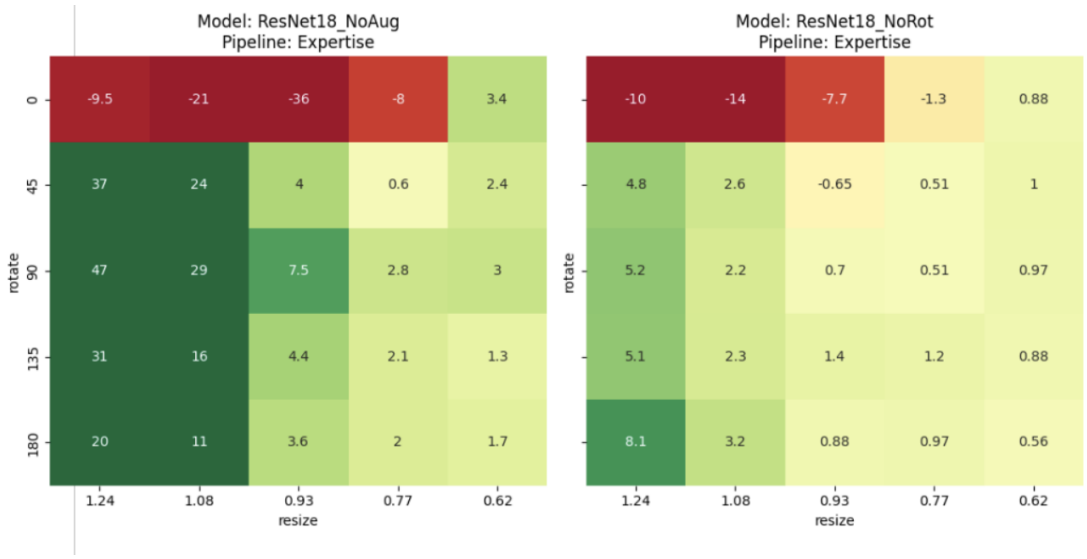

Figure 4.1. Accuracy Heatmap for curriculum learning

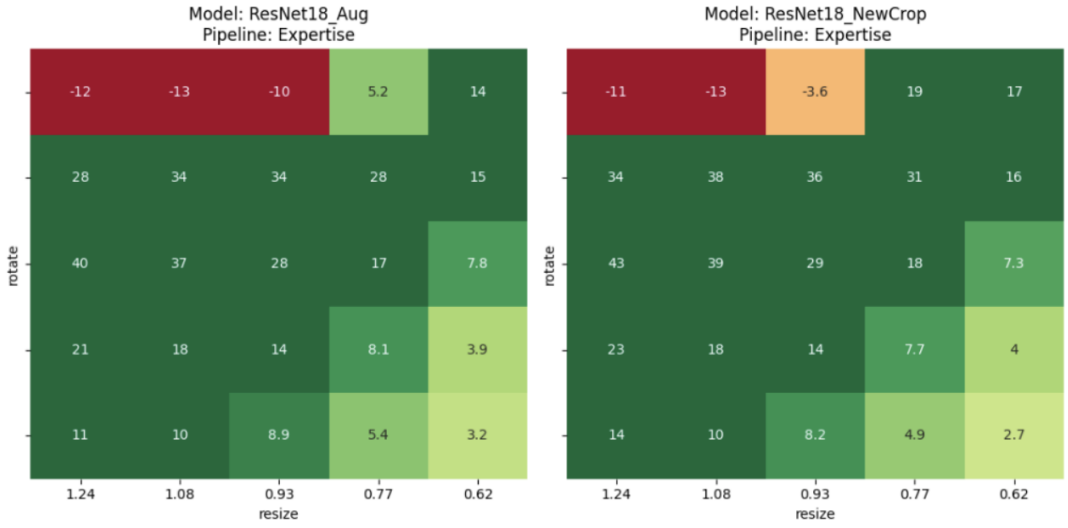

Figure 4.2. Accuracy Heatmap for curriculum learning

2. Without any augmentation in either case.

3. Using random resize crops for the Euclidean model and fixation-based augmentation for the Log Polar network.

4. Employing random non-resized crops for the Euclidean model and fixation-based augmentation for the Log Polar network.

We observe a similar trend here as in our earlier experiments: the Log Polar network demonstrates better performance when evaluated on rotated/scaled images, but not otherwise. Data augmentation significantly enhances the performance of the Log Polar network when tested on rotated/scaled images.

It is also noteworthy that, even though the model trained without augmentation comes very close to the Euclidean network during the training schedule when evaluated in the default configuration, the model trained with augmentation proves to be the overall better performer in terms of robustness.

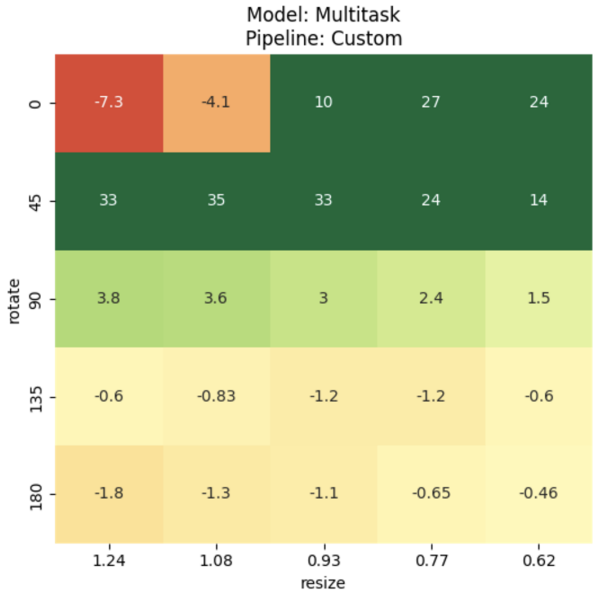

Figure 4.3. Accuracy Heatmap for Multitask learning

We observe a similar trend in the heatmap for our multitask network based on the VGG16 backbone, shown in Figure 4.3. The results closely resemble those of the vanilla VGG16 network, indicating that training the model with two losses does not significantly alter its feature representations for classification.

4.2 Noise Robustness

We also conducted a noise-robustness evaluation similar to that by Hendrycks et al. [11] to determine whether incorporating the primate log-polar mapping into network learning enhances robustness against adversarial and environmental perturbations. This evaluation considered the 19 types of noise discussed in their paper, including Gaussian noise, shot noise, impulse noise, defocus blur, motion blur, zoom blur, brightness, contrast among others. Some of the example noises in the paper are illustrated in Figure 4.4. The results of the noise-robustness evaluation for Mutitask network is presented in 4.5 and those for Curriculum models are presented in Figures 4.6, 4.7, 4.8 and 4.9, which compare the accuracy difference between Log-Polar and Euclidean models across varying levels of noise severity.

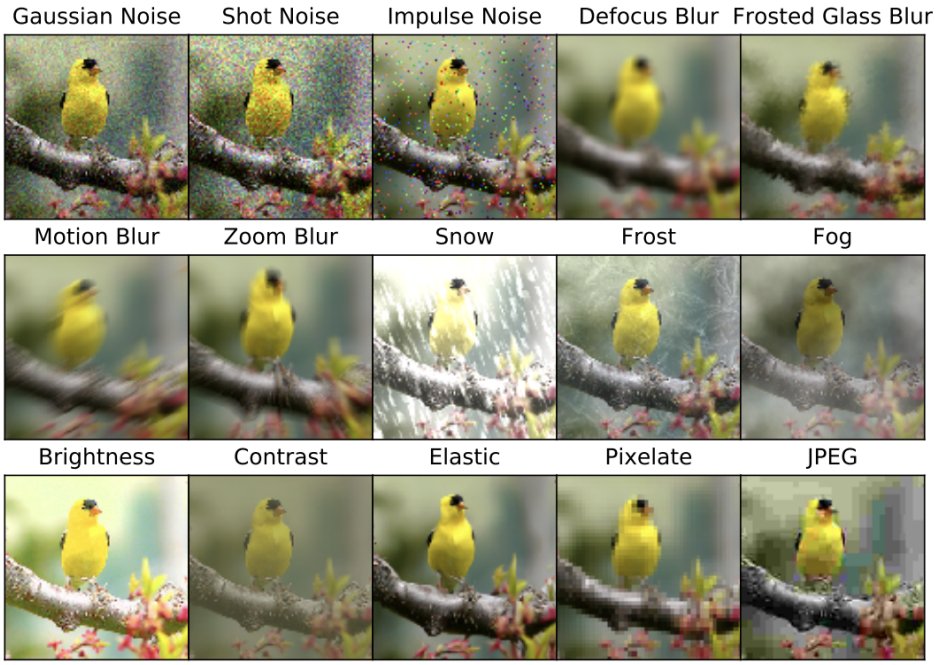

Figure 4.4. Examples of noise types considered in the noise-robustness evaluation. Source: Hendrycks Dietterich (2019) [11]

In these heatmaps, the horizontal axis represents the 19 types of noise. The 'No Noise' and '0' severity point on the x-axis indicates standard setting, which corresponds to the top-left cell in the previous set of heatmaps. The vertical axis denotes noise severity, ranging from 0 (no

distortion) at the top to 5 (extreme distortion) at the bottom.

Although the heatmap for Multitask networks does not show any clear advantage of log polar model over euclidean, the curriculum learning heatmaps reveal some really interesting patterns. We observe that, without any explicit training for noise robustness, the Log-Polar models consistently outperform Euclidean models for Gaussian noise, Shot noise, Impulse noise, and Speckle noise, demonstrating enhanced resilience across all severity levels. This suggests that the inherent properties of the log-polar transformation, such as its focus on central regions and reduction of peripheral distortions, contribute to its robustness against these types of pixel-level perturbations. For some other noise types, such as Gaussian blur and spatter, the Log-Polar models show superior performance only at higher severity levels.

On the other hand, the Euclidean models maintained an edge over Log-Polar models for motion blur, glass blur, zoom blur, elastic transform, pixelation, and certain other image-wide distortions. These types of noise, which involve global or larger-scale distortions across the image, may not align as well with the advantages provided by log-polar transformations. Log-Polar mappings are more effective in situations where local or peripheral distortions are the primary concern, but for global distortions like blur and pixelation, the traditional Euclidean space might offer better preservation of overall image structure and content, bypassing the localized advantages of log-polar mappings.

These results suggest that the log-polar mapping may be particularly well-suited for addressing noise types that involve localized, high-frequency perturbations, rather than global distortions. Further investigation into this phenomenon could provide insights into how biologically inspired transformations can be optimized for adversarial robustness.

For a complete description of the noise types and their characteristics, readers are encouraged to refer to Hendrycks et al. [11].

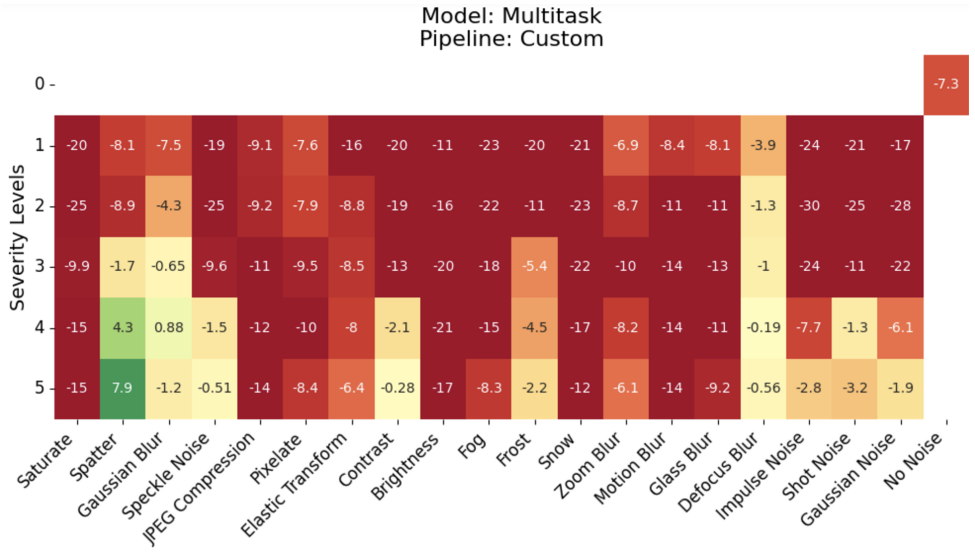

Figure 4.5. Accuracy Heatmap for Multitask Network - Noise Robustness

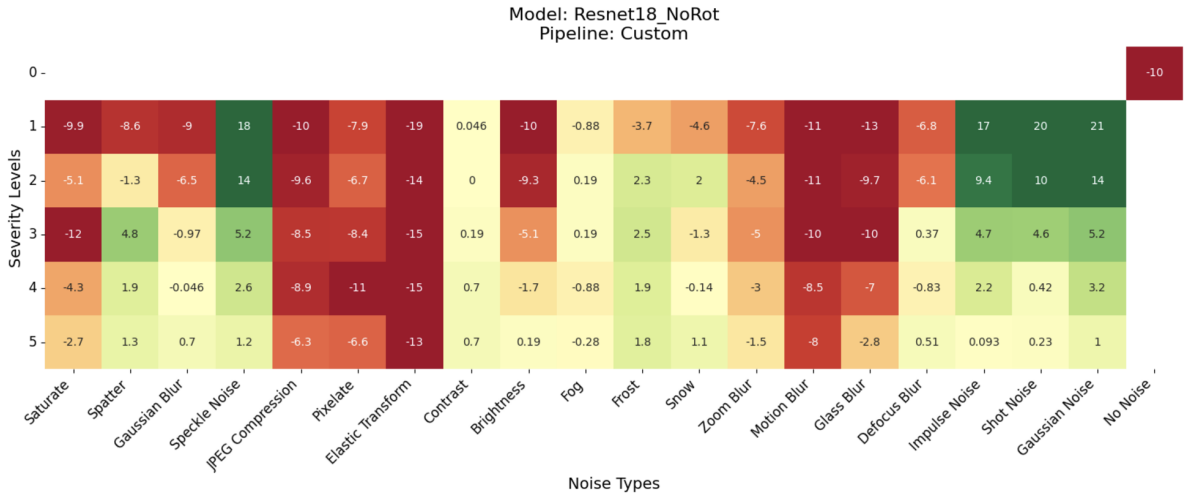

Figure 4.6. Accuracy Heatmap for curriculum learning - Noise Robustness

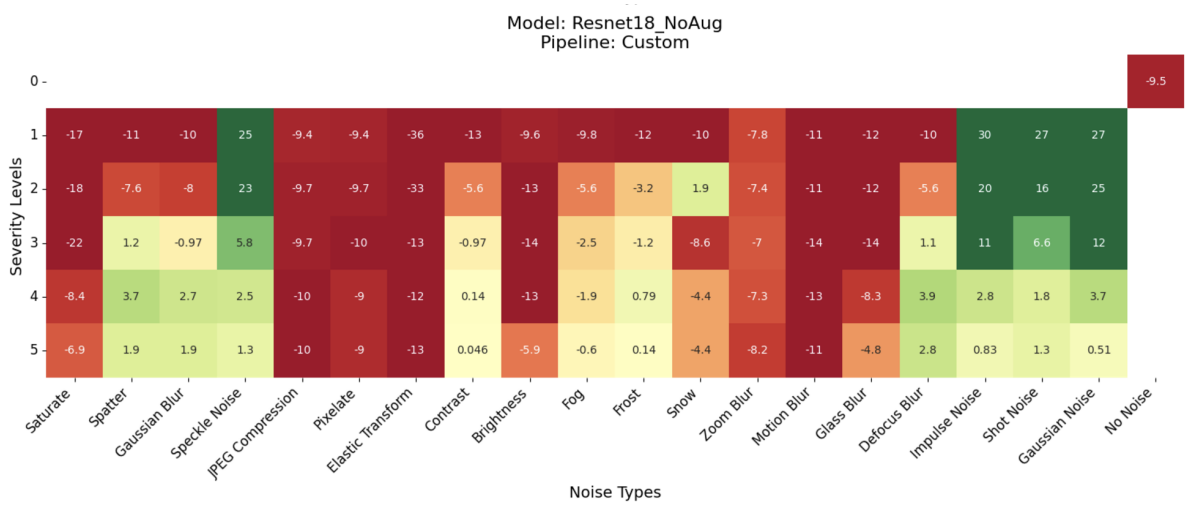

Figure 4.7. Accuracy Heatmap for curriculum learning - Noise Robustness

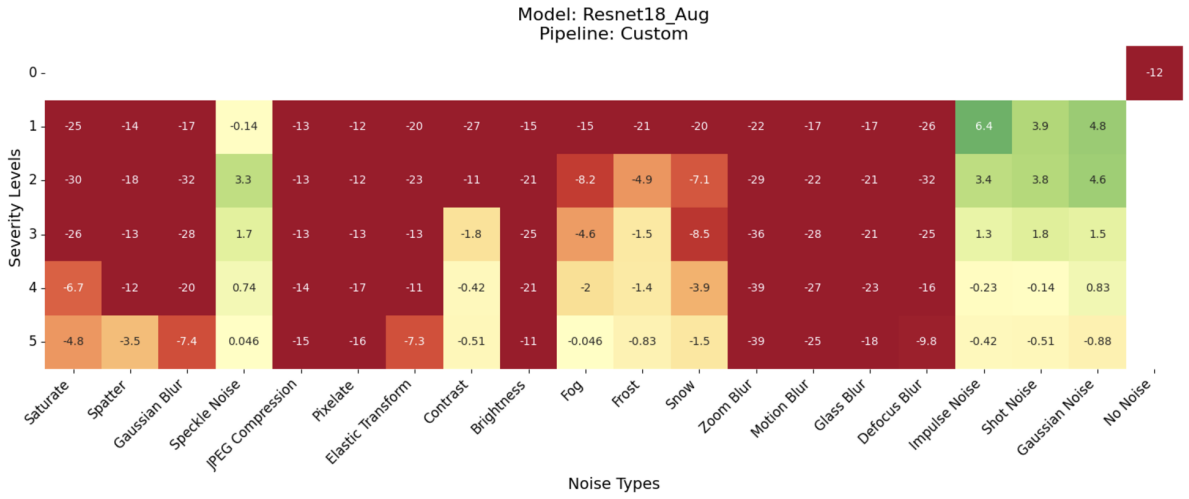

Figure 4.8. Accuracy Heatmap for curriculum learning - Noise Robustness

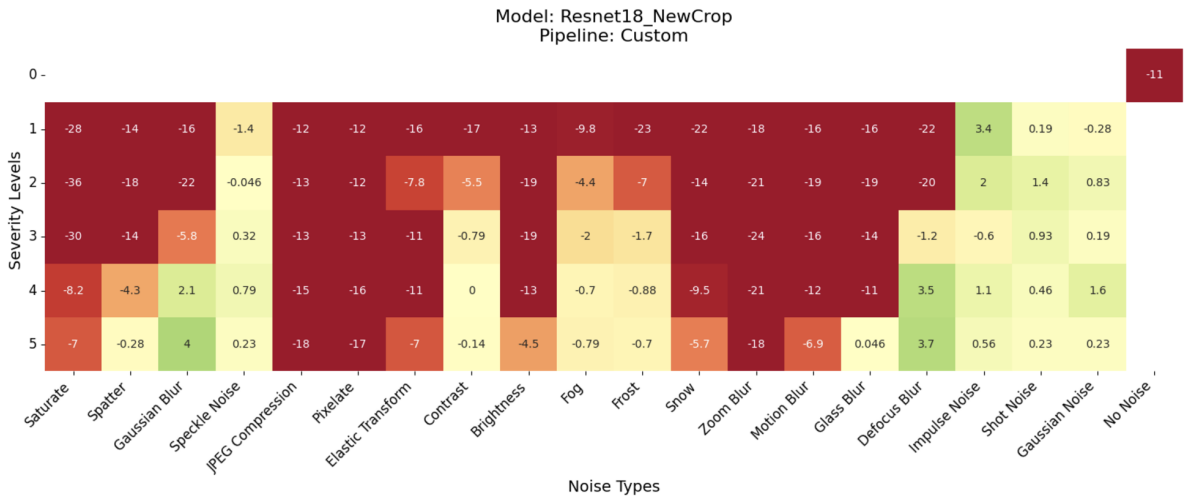

Figure 4.9. Accuracy Heatmap for curriculum learning - Noise Robustness

Chapter 5

Conclusion

This study highlights the advantages and limitations of employing log-polar transformations in conjunction with convolutional neural networks (CNNs). By mapping images into log-polar space, the models gain improved robustness to rotation and scale changes, which aligns with certain properties of the human visual system. Through a series of experiments on facial recognition tasks, we observed that while log-polar networks excel under rotated and scaled conditions, they still struggle to outperform Euclidean counterparts in standard scenarios.

Our exploration of multitask learning, saliency-based fixation points, and curriculum learning investigated potential paths to bridge this performance gap. Although no combination we tried consistently yielded high performance in standard scenarios, these methods, especially curriculum learning, revealed interesting trends in evaluation, particularly under variations in noise and orientation.

Interestingly, log-polar networks demonstrated unexpected robustness to specific types of noise, such as Gaussian, shot, impulse, and speckle noise, even without explicit training for these perturbations. This finding suggests potential for further exploration into other robustness that it might offer by virtue of the inherent characteristics of log-polar transformations. Expanding this analysis to include additional noise types, such as compression artifacts or spatial distortions, could provide a deeper understanding of the transformation's benefits. Moreover, this could reveal new ways to leverage log-polar networks for applications requiring robust performance

under adverse imaging conditions.

Additionally, log-polar networks could be explored for modeling other neurological phenomena, particularly those related to varying image orientations, such as the face inversion effect by Yin in [24], which demonstrated that recognizing inverted faces is significantly harder compared to upright ones, highlighting the unique processing mechanisms for faces in human vision. Given the alignment of these networks with properties of the human visual system, investigating their performance on inverted faces, shape vs texture bias, and other perceptual phenomena could help evaluate their applicability for tasks that require human-like perception.

Bibliography

[1] Helder Araujo and Jorge Dias. An introduction to the log-polar mapping [image sampling]. In *Proceedings of the 2nd IEEE Workshop on Applications of Computer Vision*, pages 139–144, January 1997.

[2] Nicholas Baker and James H. Elder. Deep learning models fail to capture the configural nature of human shape perception. *iScience*, 25(9):104913, 2022.

[3] Yoshua Bengio, Jérôme Louradour, Ronan Collobert, and Jason Weston. Curriculum learning. In *International Conference on Machine Learning*, 2009.

[4] Jie Cao, Chun Bao, Qun Hao, Yang Cheng, and Chenglin Chen. Lpnet: Retina inspired neural network for object detection and recognition. *Electronics*, 10(22), 2021.

[5] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey E. Hinton. A simple framework for contrastive learning of visual representations. *ArXiv*, abs/2002.05709, 2020.

[6] Carol L. Colby and Michael E. Goldberg. Space and attention in parietal cortex. *Annual review of neuroscience*, 22:319–49, 1999.

[7] Jeffrey L. Elman. Learning and development in neural networks: the importance of starting small. *Cognition*, 48:71–99, 1993.

[8] Martha Gahl, Shubham Kulkarni, Nikhil Pathak, Alex Russell, and Garrison W. Cottrell. Visual expertise explains image inversion effects. In *UniReps: the First Workshop on Unifying Representations in Neural Models*, 2023.

[9] Martha Gahl, Meilu Yuan, Arun Sugumar, and Garrison Cottrell. The face inversion effect and the anatomical mapping from the visual field to the primary visual cortex. In *Proceedings of the 42nd Annual Conference of the Cognitive Science Society*, 2020.

[10] Ian J. Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial examples. *CoRR*, abs/1412.6572, 2014.

[11] Dan Hendrycks and Thomas G. Dietterich. Benchmarking neural network robustness to common corruptions and perturbations. *CoRR*, abs/1903.12261, 2019.

[12] Xiaodi Hou and Liqing Zhang. Saliency detection: A spectral residual approach. *2007 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–8, 2007.

[13] Omisa Jinsi, Margaret M. Henderson, and Michael J. Tarr. Early experience with low-pass filtered images facilitates visual category learning in a neural network model. *PLOS ONE*, 18(1):1–25, 01 2023.

[14] James L McClelland and Timothy T Rogers. The parallel distributed processing approach to semantic cognition. *Nature Reviews Neuroscience*, 4(4):310–322, 2003.

[15] Massimiliano Patacchiola and Angelo Cangelosi. Head pose estimation in the wild using convolutional neural networks and adaptive gradient methods. *Pattern Recognit.*, 71:132–143, 2017.

[16] Jeffrey S. Perry and Wilson S. Geisler. Gaze-contingent real-time simulation of arbitrary visual fields. *electronic imaging*, 4662:57–69, 2002.

[17] J. Polimeni, Mukund Balasubramanian, and Elaine Schwartz. Multi-area visuotopic map complexes in macaque striate and extra-striate cortex. *Vision Research*, 46:3336–3359, 2006.

[18] Leendert A. Remmelzwaal, Amit Kumar Mishra, and George F. R. Ellis. Human eye inspired log-polar pre-processing for neural networks, 2019.

[19] Jignesh N. Sarvaiya, Suprava Patnaik, and Salman R. Bombaywala. Image registration using log-polar transform and phase correlation. pages 1–5, 2009.

[20] Bing Su and Ji rong Wen. Log-polar space convolution for convolutional neural networks. *ArXiv*, abs/2107.11943, 2021.

[21] Christian Szegedy, Wojciech Zaremba, Ilya Sutskever, Joan Bruna, D. Erhan, Ian J. Goodfellow, and Rob Fergus. Intriguing properties of neural networks. *CoRR*, abs/1312.6199, 2013.

[22] Saikiran S. Thunuguntla. Object tracking using log-polar transformation. In *LSU Master's Theses. 4238.*, 2005.

[23] Stéfan van der Walt, Johannes L. Schönberger, Juan Nunez-Iglesias, François Boulogne, Joshua D. Warner, Neil Yager, Emmanuelle Gouillart, and Tony Yu. scikit-image: Image processing in python. *CoRR*, abs/1407.6245, 2014.

[24] Robert K. Yin. Looking at upside-down faces. *Journal of Experimental Psychology*, 81(1):141–145, 1969.