Fully Complex Valued Convolutional Neural Network for Keypoints Detection

Kollamoram Karthik (2430322)   Dr Koteswar Rao Jerripothula

I. Abstract

Keypoints [1], also known as feature points, are specific, identifiable points within an image, such as facial landmarks or object corners, which can be used to make interpretations about the overall structure and features of the image. These keypoints are essential for various applications, including facial recognition, pose estimation, and object tracking. They find many applications industrially, such as in healthcare for patient monitoring, in robotics for precise movement, and in augmented reality for immersive experiences.

Despite significant advancements in using convolutional neural networks (CNNs) for keypoints detection [2], the complex-valued domain remains underexplored. This project aims to address this gap by developing a model inspired by Fully Complex-valued Convolutional Networks (FCCNs) [3]. Our approach involves creating complex-valued inputs, processing them with complex convolutions, and representing outputs as complex numbers to maintain the flow of complex information throughout the network. By leveraging these complex-valued operations, our project seeks to enhance the accuracy of keypoints detection systems, setting the stage for future research in this promising area.

II. Introduction

Convolutional Neural Networks (CNNs) have shown remarkable performance in keypoint detection tasks, yet they face some limitations when it comes to handling complex-valued data. The challenges include but are not limited to: (i) Converting real-valued image data into complex-valued inputs. (ii) Processing these inputs using standard convolutions that are not equipped to handle complex-valued operations. (iii) Representing keypoints as complex valued outputs to maintain the flow of complex information. (iv) Ensuring continuous complex information flow despite the disruption caused by fully connected layers.

To address these challenges, our project aims to (i) create complex-valued inputs from real-valued images using the Complex-valued Local Binary Pattern (iLBP) function. (ii) process these inputs with complex convolutions from the Complex Keras library [4], (iii) represent keypoints as complex numbers (x + iy) to maintain complex information flow, and (iv) replace fully connected layers with 1x1 complex convolutions and global average pooling to ensure continuous complex information flow and reduce model parameters. By solving these problems, our project enhances keypoint detection through an efficient model inspired by FCCNs

III. Data Set

The dataset used in this project is from the Kaggle Facial Keypoints Detection competition [6]. It consists of 7049 training images and 1783 test images, each of which is grayscale and sized at a 96x96 resolution. The keypoints in this dataset are represented as (x, y) coordinates, corresponding to specific facial features.

with keypoints, but there is an uneven distribution of keypoints per image. Specifically, 2140 images are annotated with 15 keypoints each, while the remaining 4909 images have only 4 keypoints per image. The 15 keypoints correspond to various facial features, including the eyes, nose, and mouth, which are critical for detailed facial landmark detection. The images with 4 keypoints typically focus on broader features and are less detailed.

For the purposes of this project, we trained our models on the subset of images annotated with 15 keypoints to leverage the richer annotations for more accurate keypoint detection. The test images, while maintaining the same 96x96 grayscale format as the training images, do not include keypoint annotations, which is typical for a machine learning competition where the goal is to predict these keypoints.

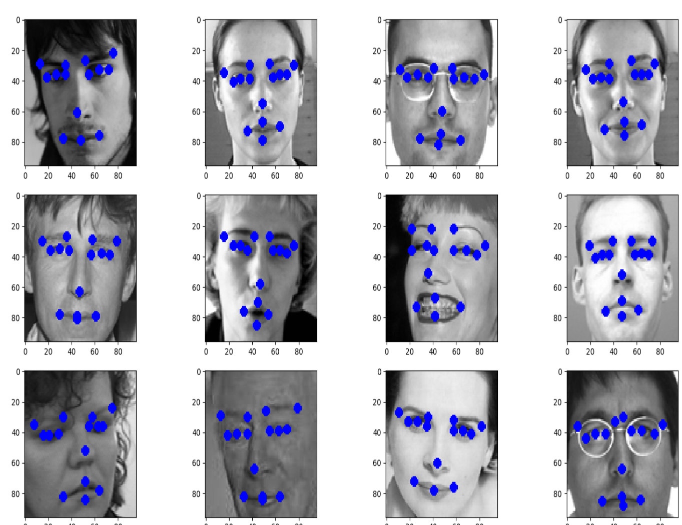


Fig. 1: Examples of the original ground-truth images and their keypoints

IV. Study Methodology

Our research methodology followed a structured approach to implement and evaluate a complex-valued neural network for keypoint detection:

1) **Implementation & Training of a traditional CNN:** The project commenced with the development of a conventional Convolutional Neural Network (CNN) for keypoint detection, establishing our baseline model.
2) **Implementation of an iCNN using Complex Keras:** We transitioned to implementing an initial Complex-valued CNN (iCNN) utilizing Complex Keras, incorporating complex convolutions and complex batch normalization layers.
3) **Analysis using augmented inputs:** To investigate the network's internal dynamics, we augmented our real-valued inputs and outputs with zero-valued imaginary components. This allowed us to visualize and analyze the intermediate layer outputs, providing insights into the model's complex data processing.
4) **Optimizing network architecture:** We replaced fully connected layers with complex 1x1 convolutional layers and global average pooling. This modification maintained the complex information flow throughout the network while reducing the overall parameter count.
5) **Implementing complex-valued input representation:** We employed the existing Complex-valued Local Binary Pattern (iLBP) function to transform real-valued image data into complex-valued inputs, enhancing the model's ability to process complex information from the initial stages.
6) **Restructuring keypoint data:** The keypoint location data was reformatted to represent x and y coordinates as real and imaginary components, fully leveraging complex number representation.
7) **Finalizing and training the complex-valued model:** With the complex-valued architecture and data representations in place, we completed the implementation and training of our final model for keypoint detection.
8) **Performance evaluation:** We conducted a concise comparison between our complex-valued model and the traditional CNN, examining key performance metrics.

This methodical approach facilitated the systematic development and evaluation of our complex-valued model for keypoint detection.

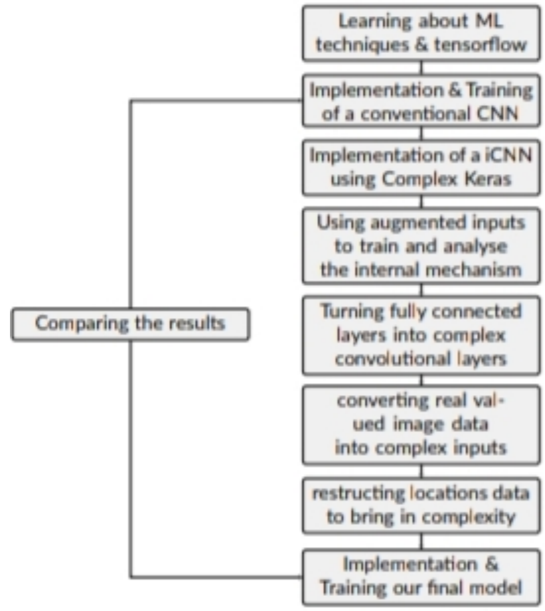


Fig. 2: Project Progress Flow Chart

V. How did we convert real valued images into complex inputs?

To transform real-valued image data into complex inputs, we employ the Complex-valued Local Binary Pattern (iLBP) function. This method involves shifting the image in eight directions, calculating pixel-wise differences, and multiplying these by unit complex numbers corresponding to each direction. The resulting complex representation consists of real (FCR) and imaginary (FCI) sub-images, capturing both magnitude and phase information from the original image.

The iLBP function excels in capturing fine texture details by integrating phase information with intensity variations. This approach enhances our ability to analyze and process images in scenarios where traditional methods focusing solely on intensity may not suffice. By leveraging iLBP, our project achieves a more nuanced understanding of image textures, facilitating advanced image processing tasks that require a comprehensive analysis of both intensity and phase characteristics.

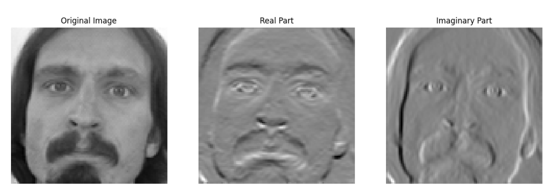


Fig. 3: An image with its sub images generated by the iLBP function

VI. Model Architecture and Functioning

Our neural network for keypoint detection is built using a series of complex-valued convolutional layers, each followed by LeakyReLU activation and ComplexBatchNormalization. MaxPool2D layers are used for downsampling after each block. The architecture progresses through six complex convolutional blocks, with filter sizes increasing from 16 to 512.

A 1x1 ComplexConv2D layer reduces the depth of the feature maps, followed by a Dropout layer to prevent overfitting. Another 1x1 ComplexConv2D layer produces the final keypoint predictions, and a GlobalAveragePooling2D layer aggregates these predictions. The model employs the Adam optimizer and Mean Squared Error loss function to train effectively.

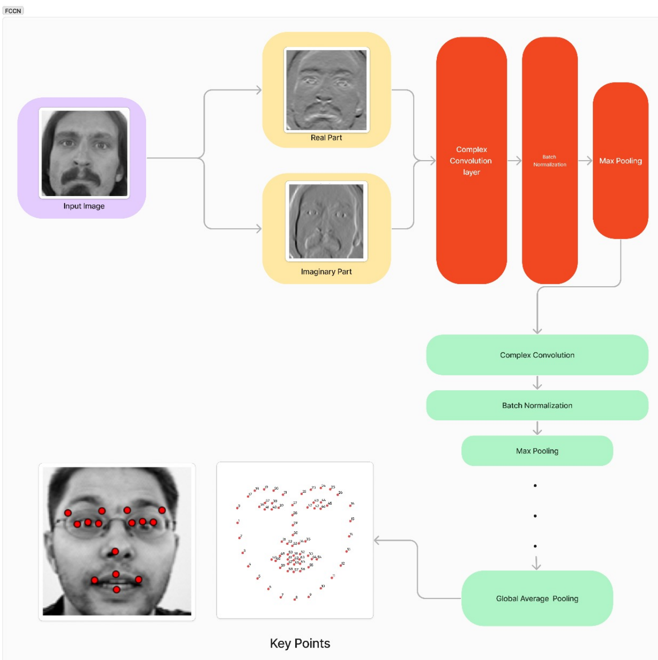


Fig. 4: Neural Network Architecture

VII. Results and Discussion

Our model, inspired by concepts from Fully Complex Valued Convolutional Networks (FCCNs), demonstrated notable improvements in keypoint detection compared to a traditional Convolutional Neural Network (CNN) of similar architecture. Despite having significantly fewer parameters, our implemented model achieved higher performance across all measured metrics, including validation accuracy, precision, recall, and F1 score.

The visual examples provided in the Fig. 5 illustrates the enhanced ability of our model to accurately locate facial landmarks compared to the traditional CNN. Our model's keypoint predictions appear more precise and closely aligned with actual facial features, particularly in challenging areas such as the corners of the eyes and mouth.

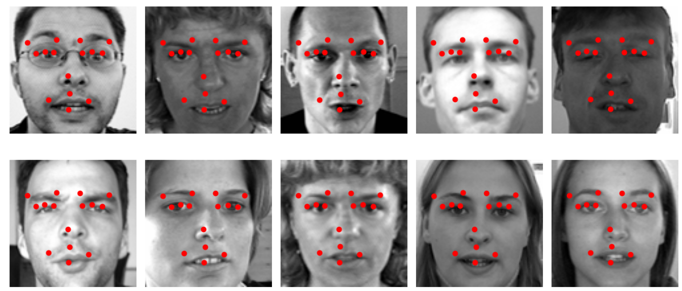
This performance improvement, achieved with a more compact model architecture, highlights the potential of leveraging complex-valued operations in neural networks for keypoint detection tasks. Our results suggest that incorporating FCCN-inspired concepts can lead to more efficient and effective models in computer vision applications, particularly those requiring precise feature localization.

VIII. Conclusion

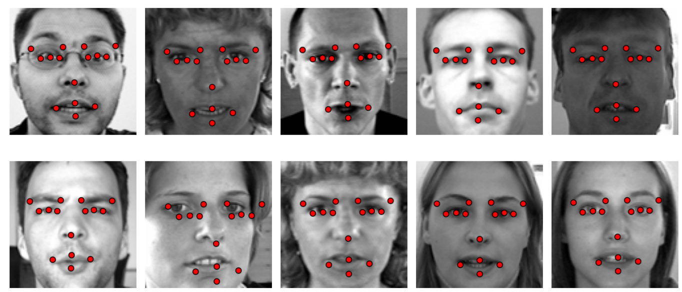
Our study demonstrates the efficency of integrating complex-valued operations in keypoint detection, inspired by Fully Complex-valued Convolutional Networks. The implemented model achieved superior accuracy compared to a traditional CNN while utilizing fewer parameters. This enhancement in performance, coupled with reduced computational demands, underscores the potential efficiency gains in optimizing neural architectures for complex information processing. Our findings suggest a promising avenue for future research in computer vision, particularly in developing more efficient and accurate models for keypoint detection tasks. This work sets a foundation for future advancements in handling complex visual tasks effectively.

TABLE I: Compiled Results

Metrics	CNN	Our Model
Model size (MB)	27.73	12.54
F1 Score	0.596	0.647
Precision	0.511	0.723
Validation Accuracy	0.715	0.734



(a) Keypoints detected by CNN (Total params: 7268670 (27.73 MB))



(b) Keypoints detected by FCCN (Total params: 3,288,222 (12.54 MB))

Fig. 5: Comparison of keypoints detected by a conventional CNN and our complex valued convolutional neural network

IX. Future Prospects

1) Model Optimization: Potential for developing custom complex loss functions and activation functions, opening new avenues in complex number analysis for machine learning.
2) Industrial Applications: Our approach, achieving better results with less computational effort, has broad potential across industries using keypoint detection:
   a) Healthcare: Improving patient monitoring systems.
   b) Robotics: Enhancing precise movement tracking.

c) Augmented Reality: Advancing immersive experiences.

X. REFERENCES

[1] Stack Overflow. (2015). What are keypoints in image processing? Retrieved from [link]

[2] Potrimba, P. (2023, October 31). What is Keypoint Detection? Roboflow Blog. Retrieved from [link]

[3] Saurabh Yadav and Koteswar Rao Jerripothula,"FCCNs: Fully Complex-valued Convolutional Networks using Complex-valued Color Model and Loss Function," in IEEE/CVF International Conference on Computer Vision (ICCV), 2023. [Online]. Available: [pdf] [code]

[4] Jesper Sören, Dramsch and Contributors, "Complex-Valued Neural Networks in Keras with Tensorflow," figshare, 2019. [Online]. Available: [link]. DOI: 10.6084/m9.figshare.9783773.

[5] Chiheb Trabelsi, Olexa Bilaniuk, Ying Zhang, Dmitriy Serdyuk, Sandeep Subramanian, João Felipe Santos, Soroush Mehri, Negar Rostamzadeh, Yoshua Bengio, Christopher J. Pal, "Deep Complex Networks," arXiv preprint arXiv:1705.09792, 2017.

[6] hNao17. (2024). facial-keypoints-detection. GitHub repository. Retrieved from [link].

[6] Kaggle, "Facial Keypoints Detection," Kaggle Competition, [Online]. Available: link

Koteswar Rao

Mentor's Signature:
Dr. Koteshwar Rao Jerripothula,
Department of Electrical Engineering,
Indian Institute of Technology Kanpur.