|  |  |  |  |
| --- | --- | --- | --- |
| Roll No | 21BCP359 | Practical No: | 6 |
| Division | 6 | Date: | 13/03/2024 |

|  |
| --- |
| Part 1  Understand the project available on following link  Project Link: <https://github.com/aharley/nn_vis>  Project by: <https://adamharley.com/>  Reference in case needed: <https://www.youtube.com/watch?v=pj9-rr1wDhM> |
| Part 2  **Populate the table below to summarize your understanding of the project mentioned in part 1**   |  |  |  | | --- | --- | --- | | Layer | Task | Rationale | | Input layer | It receives user input and converts raw pixels from a sketchpad into data that the system can process. | To take raw input from the user and preprocess it. | | Convolutional layer | It identifies patterns in the input data, like edges and corners, by applying mathematical operations and activation functions. | Extracts features, like edges and corners. | | Pooling layer | It reduces the size of the data while keeping important information intact, making computations more efficient. It does this by condensing features and focusing on the most significant values. | To reduce space of the matrix (reduce spatial dimensions of feature maps) while conserving the original image. Consider the pixel having the highest value (illumination) using a stride of 2\*2 pixels (2\*2 max pooling) and taking that value to just one pixel in the new matrix. | | Classifying layer | It utilizes the extracted features to accurately categorize the input data. It consists of interconnected neurons that analyze the features for classification. | The classifying layer takes the high-level abstracted features from previous layers and uses them to classify input data into different categories. There are 120 neurons in the first layer and 100 neurons in the second. | | Output layer | It generates the final prediction based on the classification results, with each neuron representing the probability of a specific outcome, such as recognizing different digits | Produces the final output or prediction of the network, representing the class probabilities | |  |  |  | |
|  |
| **How does the following hyper-parameters affect the network performance**   |  |  |  | | --- | --- | --- | | **Hyper-Parameter** | **One Line Definition** | **Effect on the CNN** | | Stride | Determines how much the filter moves across the input image. | Changing the stride impacts the size of the output feature maps. A larger stride means fewer calculations and smaller output maps, speeding up processing. | | Dilation Rate | Controls how the elements of the convolutional filter are spread out. | Increasing dilation rate expands the filter's view, allowing it to capture broader features but at the cost of reduced detail in the output. | | Type of pooling layer | Dictates how feature maps are condensed in pooling layers. | Various types like max pooling or average pooling determine how features are combined, impacting the network's capacity to maintain crucial details while decreasing size. | | Kernel size | Determines the dimensions of the convolutional filters. | Bigger sizes gather more nearby details, enabling the network to learn complex patterns and demanding more computations. Smaller sizes concentrate on finer details but might miss broader patterns. | | padding | Adding extra pixels around the input image. | It influences the size of the output feature maps. Zeropadding keeps the size unchanged, valid padding reduces it, and same padding maintains the input size. | |
|  |
| References:  [An Intuitive Explanation of Convolutional Neural Networks – the data science blog (ujjwalkarn.me)](https://ujjwalkarn.me/2016/08/11/intuitive-explanation-convnets/)  [Gentle Dive into Math Behind Convolutional Neural Networks | by Piotr Skalski | Towards Data Science](https://towardsdatascience.com/gentle-dive-into-math-behind-convolutional-neural-networks-79a07dd44cf9)  [Intuitively Understanding Convolutions for Deep Learning | by Irhum Shafkat | Towards Data Science](https://towardsdatascience.com/intuitively-understanding-convolutions-for-deep-learning-1f6f42faee1)  [An Introduction to different Types of Convolutions in Deep Learning | by Paul-Louis Pröve | Towards Data Science](https://towardsdatascience.com/types-of-convolutions-in-deep-learning-717013397f4d) |