Skin Cancer Classification Using CNN

Skin cancer is a type of cancer that affects the skin cells. It is the most common form of cancer in the United States. There are three main types of skin cancer: basal cell carcinoma, squamous cell carcinoma, and melanoma. Basal cell carcinoma and squamous cell carcinoma are the most common types and are usually not life-threatening, but melanoma can be aggressive and spread to other parts of the body.

Risk factors for skin cancer include excessive sun exposure, sunburns, fair skin, family history, and a weakened immune system. Symptoms can include a new or changing mole, a sore that doesn't heal, or a bump or spot that's growing.

Diagnosis is typically made through a skin biopsy, in which a sample of skin is removed and examined under a microscope. Treatment options depend on the type and stage of the cancer and may include surgery, radiation therapy, or chemotherapy. Prevention measures include avoiding sun exposure, wearing protective clothing and using a broad-spectrum sunscreen with an SPF of at least 30. Early detection is crucial for successful treatment, so it's important to have regular skin checks and be aware of any changes to your skin.

Dataset Overview

HAM10000 ("Human Against Machine with 10000 training images") dataset - a large collection of multi-source dermatoscopic images of pigmented lesions

The dermatoscopic images are collected from different populations, acquired and stored by different modalities. The final dataset consists of 10015 dermatoscopic images.

It has 7 different classes of skin cancer which are listed below :

1. Melanocytic nevi
2. Melanoma
3. Benign keratosis-like lesions
4. Basal cell carcinoma
5. Actinic keratoses
6. Vascular lesions

7. Dermatofibroma

CODE CONTENTS:

- Importing libraries
- Reading the Data
- Data Preprocessing
- Exploratory Data Analysis
- Data Modeling
- Accuracy of model
- Plotting Confusion Matrices

Design:

- The model starts with a sequential architecture, which allows for the sequential stacking of layers.
- It consists of multiple layers, including Conv2D (convolutional), MaxPool2D (max pooling), Dropout (regularization), and Dense (fully connected) layers.
- The model uses the ReLU activation function for the convolutional and dense layers, except for the last layer which uses softmax activation for multi-class classification.
- Adam optimizer is used with a learning rate of 0.001 and other specified parameters.
- The model uses the categorical cross-entropy loss function for multi-class classification.
- Learning rate reduction is applied using the ReduceLROnPlateau callback, which reduces the learning rate when the validation accuracy plateaus.

Approach:

- The model follows a typical CNN architecture commonly used for image classification tasks.
- It starts with a pair of Conv2D layers with 32 filters each, followed by a MaxPool2D layer for downsampling and a Dropout layer for regularization.
- This is followed by another pair of Conv2D layers with 64 filters each, another MaxPool2D layer, and a higher dropout rate.
- The output from the convolutional layers is then flattened and passed through a fully connected Dense layer with 128 units and ReLU activation, followed by another Dropout layer.
- Finally, the output is passed through a Dense layer with the number of units equal to the number of classes in the classification problem, using softmax activation for multi-class classification.

Reasoning:

- Convolutional layers capture spatial hierarchies and local patterns in images, making them suitable for image classification tasks.
- Max pooling layers downsample the spatial dimensions, reducing computational complexity and providing translation invariance.
- Dropout layers help in regularizing the model and preventing overfitting by randomly dropping out a fraction of the units during training.
- The choice of the ReLU activation function is common for CNNs as it introduces non-linearity and speeds up training.
- The softmax activation in the last layer converts the model's output into probabilities, making it suitable for multi-class classification problems.
- Adam optimizer is a popular choice for training neural networks due to its adaptive learning rate and momentum properties.

- Learning rate reduction helps in fine-tuning the learning rate during training, which can improve convergence and accuracy.

Overall, the code implements a CNN model with data augmentation (using ImageDataGenerator) to prevent overfitting. The model is trained using the fit_generator function, which takes batches of augmented data during training. The model's performance is monitored using validation data, and the learning rate is reduced if the validation accuracy plateaus. Finally, the model's architecture is visualized using the plot_model function.