Deep Supermarket: Transfer Learning Approach for Classification of Indian Supermarket Products

Varun Verma, Dr. Noel Jeygar Robert V
Vellore Institute of Technology, Chennai
Kelambakkam - Vandalur Rd, Rajan Nagar, Chennai, India
varun.verma@vitstudent.ac.in, noeljeygar.robert@vit.ac.in

ABSTRACT

To identify an optimal DNN model and approach to classify images of Indian supermarket products this study will undertake a comparative evaluation of different deep neural network models and techniques. Both traditional and transfer learning (TL) techniques will be used for this purpose. There was a lack of a proper image dataset specifically for Indian supermarket and grocery products identified, hence necessitating the creation of a unique image dataset (Indian Grocery Image dataset_v3). At first, Convolutional Neural Network (CNN) and VGG19 architectures were developed but their performance was suboptimal. To improve the classification performance Transfer Learning methods were used by employing pre-trained models such as EfficientNetB7, InceptionResNetV2, DenseNet169, and DenseNet201 which had been trained on the ImageNet dataset. The traditional DNN models were outperformed by the transfer learning models with exceptional performance demonstrated by InceptionResNetV2 and the DenseNet family of DNNs. Among them, the DenseNet201 model performed the best among the models under study. With training accuracy being 99.36% and validation accuracy being 80.47%, therefore making it most optimal among the architectures for our problem statement of classifying Indian supermarket products in this research.

Keywords: *Transfer Learning, CNN, VGG19, EfficientNetB7, InceptionResNetV2, DenseNet169, DenseNet201, Indian Supermarket Products, Custom Dataset*

1. INTRODUCTION

In the era of automation, places like grocery stores, supermarkets, and warehouses are moving towards more autonomous solutions like self-checkouts and automated self-categorizing and organizing warehouses. These technologies have improved the convenience and reduced the time for the consumers. These technologies run on Deep neural networks and require a good classifier to identify the product from images. Hence there is a demand for efficient DNN models that can categorize these products in these retail environments. Especially there is a gap of such models for the Indian supermarket landscape. As these models need to be trained on a well-oriented dataset, there was extensive research on image classifiers for foreign products (belonging to countries like the US, UK, and UAE) however there has been very limited investigation into classifiers for Indian grocery/supermarket goods.

The research aims to fill this void. It examines the application of DNN models to correctly identify Indian supermarket or grocery products using the Transfer Learning (TL) approach. The classifier would be trained on a self-curated dataset consisting of 37,310 images of Indian products which are categorized in 15 distinct classes. A custom dataset was required to ensure that the model was trained on the images of Indian supermarket products. Through the review of the literature and extensive online search, an absence of a pre-existing dataset was found, making the creation of a new Image dataset an integral part of the study for the proper creation of classifier models.

Deep Learning models were chosen to be evaluated for this study as they were the dominating tools when it comes to tasks like image classification, identification of objects, and computer vision. The initial approach consisted of the more traditional approach to DNN classifiers. In this approach, simple CNN models and better architectures of VGG19 models were chosen for the study's classification tasks. However despite implementing the models with the best architectural changes and providing fine-tuning, hyperparameter changes, and learning rate tweaks, both the traditional models gave unsatisfactory performance for classification, scoring poorly in training and testing accuracies.

The study then changed its direction to investigation of more powerful approaches namely the Transfer Learning approach. TL is leveraged by utilizing complex models like the EfficientNet, and DenseNet family of architectures which have been pre-trained on the ImageNet dataset (or any other large Image dataset). The models retain their knowledge gained from that pre-training and it is applied to solve another related problem. TL has revolutionized the field of DNN as it not only gives us pre-trained complex models but also decreases the computational resources and time to train significantly. Classification accuracy according to the literature was almost always higher for models that had been trained using the TL approach, compared to the traditional approach.

The EfficientNetB7 model, due to it being the Largest of the EfficientNet family of models and its highly complex architectural design was projected to show the best performance among all the models. However, it only marginally surpassed the traditional models, though it did perform better than those models. Also, it showed the least amount of overfitting, with its testing and training scores beginning almost the same. On the other hand, InceptionResNetV2 and the DenseNet family of models (DenseNet169, DenseNet201) exhibited much higher performance but also showed signs of overfitting. The research compares the different architectures and approaches' performances through empirical and graphical methods and suggests the most suitable model for the classification of Indian supermarket items among the architectures under study

Thus, the two primary challenges addressed by this research are: creating a new unique Image dataset of Indian supermarket/grocery products and identifying an efficient deep learning model for the classification of that dataset hence creating an optimal Indian supermarket product image classifier. The study uses a Transfer learning approach and a new dataset to fill in the literature gap identified and offers insights into the creation of automated models for the categorization of Indian products. This would encourage more studies and research on this topic, which would improve the retail operations and automation sector of India.

2. RELATED WORKS

• "Store product classification using CNN" by I Made Wiryana, Suryadi Harmanto, Alfharizky Fauzi, Imam Bil Qisthi, Zalita Nadya Utami [1] - To enhance the efficiency and cost-effectiveness of store product sorting, Convolutional Neural Network (CNN) designs are discussed in this paper. The study goes into deep learning-based retail product identification with an emphasis on CNN models for object detection. This paper is based on an extensive literature review which brings to light some of the challenges, methodologies, and datasets involved in deep learning-based product recognition thus providing insights that will be useful to researchers and practitioners in this field. There is a lot of detail given regarding how computer vision can develop from its traditional forms to embrace new-age techniques such as deep learning.

• Muhathir et al.'s "Convolutional Neural Network (CNN) of Resnet-50 with Inceptionv3 Architecture in X-Ray Image Classification" [2] - Advanced CNN architectures – Resnet-50 and Inceptionv3, are employed in classifying X-ray images. Inceptionv3 is a computationally efficient convent employing factorized convolutions, smaller convolutions, asymmetric convolutions, auxiliary classifiers, and grid size reduction for better network performance and reduced computational costs. Through the combination of these methods i.e., auxiliary classifiers, convolution factorization, RMSProp optimization, and Label Smoothing, it is possible to obtain Inceptionv3 which performs better than other models on ImageNet with lower error rates as observed in Christian Szegedy et al.'s 2015 paper "Rethinking the Inception Architecture for Computer Vision". The cooperation between Resnet-50 and Inceptionv3 architects demonstrates how architectural improvements can enhance the efficiency of the model as well as its accuracy concerning the classification of X-ray images.

• "An effective CNN and Transformer complementary network for medical image segmentation", ScienceDirect [3] - The article explores how medical image segmentation can be improved using CNN and Transformer encoders. CNN encoders are good at describing local features, while the Transformer encoders – capture long-term dependencies, and hence the paper decided to unify the two models. In summary, ConvFormer is a network that uses both - local information from CNN and global features from Transformer to improve medical image segmentation. ConvFormer outperforms other approaches in terms of a feed-forward module that integrates enhanced de-transformer, a hybrid stem shaped as residual to capture local and global features and an encoder of multi-scale maps. This novel approach shows how medical image analysis accuracy and efficiency can be improved using CNNs and transformers.

• "Comparison of CNN-based deep learning architectures for rice disease classification," ScienceDirect [4] – This research scrutinizes varied CNN architectures while utilizing deep learning for the classification of rice diseases. In this work, GoogleNet, ResNet-18, SqueezeNet-1.0, and DenseNet-121 are assessed in terms of their performance in detecting rice diseases. Moreover, the article points out that about identification of rice disease using CNN still has some problems to be addressed by proposing future study areas and stressing on importance of deep learning models with automatic feature extraction from images for rice disease classification. The essay compares different types of CNNs to provide insights for agricultural practitioners and researchers who are looking forward to coming up with the most appropriate DL methods to be used for classifying rice disease.

• "Deep Learning Model Based on ResNet-50 for Beef Quality Classification" S. E. Abdallah, W. M. Elmessery, M. Shams, N. SA Al-Sattary [5] - The paper employs the analysis of the surface texture to distinguish pictures of healthy and rancid beef. The number of healthy beef images was eight, while that of rancid ones was ten; therefore, a Generative Adversarial Network (GAN) was employed to include 180 more pictures. This shows deep learning's efficiency in identifying beef quality as evidenced by the ResNet-50 model which achieved 96.03% training, 91.67% testing, and 88.89% validation accuracy. Also compared with classical and other deep learning architectures, this further demonstrates the image classification efficiency of ResNet-50. For food quality evaluation purposes, an efficient robust ResNet-50-based Deep Learning Model for Beef Image Classification overcomes data limitations and improves classification accuracy through advanced deep learning methodologies.

• "Performance evaluation of ResNet model for tomato plant disease classification" by S. Kumar, Pal, Singh, and P. Jaiswal [6] - The study evaluates the ResNet model's tomato disease classification capabilities. The study tests ResNet's ability to detect tomato diseases like early and late blight, leaf mold, leaf spot, two-spotted spider mite, target spot, yellow leaf curl virus, and mosaic virus. The study compares Inception V3 and Inception ResNet V2 models using PlantVillage and other platform images of diseased and healthy tomato leaves. ResNet excels in training, testing, and validation, proving its tomato plant disease classification efficacy. This study advances plant disease classification methods and helps agricultural researchers and practitioners find the best deep-learning models for tomato disease classification.

• “An Enhanced Transfer Learning Based Classification for Skin Cancer Diagnosis” by V. Anand et al. K. Jilani Saudagar [7] - The researchers analyzed a deep learning model that is based on transfer learning for distinguishing between benign and malignant skin cancers. This work was an improvement over previous works that improved skin cancer classification by stacking and refining the VGG16 model. The methodology presented in this paper has the potential to improve classification through input dataset preparation, data augmentation, VGG16 feature extraction, and model fine-tuning. Experimental results indicate that the new approach to transfer learning not only can yield high training, testing as well as validation accuracy in skin cancer recognition but also outperforms traditional methods used for diagnosis. This research goes a long way towards making improvements to automated systems of skin cancer detection and thereby helps clinical researchers better develop diagnostic tools.

• “A Study of CNN and Transfer Learning in Medical Imaging: Benefits, Challenges, and Future Prospects” by A. Salehi et al [8] – In the study, CNNs and transfer learning are thoroughly reviewed. They emphasize how these methods improved medical image analysis accuracy, resource use, and efficiency. The paper addresses limited training data, overfitting, and transfer learning method selection. CNN components and deep learning algorithm hardware platforms are explained in detail. The study reviews current research and explores advanced CNN architectures to help medical imaging researchers and students use CNNs and transfer learning for better diagnostics and healthcare efficiency.

• “Classification of benign and malignant subtypes of breast cancer histopathology imaging using hybrid CNN-LSTM based transfer learning - BMC Medical Imaging,” by M. M. Srikantamurthy, V. P. Subramanyam Rallabandi, D. B. Dudekula, S. Natarajan, and J. Park [9] - The main objective of this study is to come up with a deep learning model that will facilitate classification of breast cancer histopathology images as either benign or malignant and prediction of four cancer subtypes. This is done by employing a hybrid CNN-LSTM model which uses transfer learning on ImageNet and fine-tuning on BreakHis containing different magnifications of benign and malignant cancer images. The hybrid CNN-LSTM model performs better compared to ResNet50 and Inception in both binary and multi-class classification tasks after assessing with various optimizers and epochs. Such an approach has the potential to enhance clinical diagnostic accuracy towards breast cancer.

• “Fusion of U-Net and CNN model for segmentation and classification of skin lesions from dermoscopy images” – ScienceDirect [10] - The study seeks to tackle the fuzzy boundaries and irregular borders by constructing a combined model. The U-Net segmentation precision is made better by manipulating the dimensions of feature maps and increasing kernels to extract nodules more accurately. After playing with the hyperparameters like epochs, batch size, and optimizers, it was found that Adam optimizer, 8 batches, and 75 epochs worked best for this model. Modifying U-Net architecture for dermoscopy skin lesion segmentation can improve dermatology diagnosis through enhancing segmentation as well as classification.

• “Product Classification in E-Commerce Sites,” A. Vivek Patra, B. R. Shambhavi, K. Sindhu, S. Balaji [11] - Based on the research, one can say that categorization of products enhances customer satisfaction and conversions on an online shopping platform. Proper classification of a catalog improves user experience, sales volume, and website search as well as improving the shopping process. Every time you place a product in the product category, it makes sure orders are created accurately and quickly hence speeding up transactions and leading to increased e-commerce sales. Structured product pages also help analyze sales performance to make informed decisions thereby enhancing operational efficiency and customer engagement. Nevertheless, this study stresses the significance of product classification about e-commerce customer loyalty, the profitability of its operations, and trade turnovers.

• “A Machine Learning-Based Autonomous Framework for Product Classification Over Cloud,” by A. Motwani, G. Bajaj, M. Arya, S. K. Sar, S. O. Manoj [12] - An introduction of a cloud-based framework for machine learning is given for the classification of products sold over e-commerce. The methodology used in this research entailed multiclass logistic regression on a cloud computing platform to enhance efficiency and accuracy in dealing with massive product volumes. This framework aimed to make e-commerce user-friendly by coming up with an independent system that can assign products into categories without any expert knowledge. With machine learning techniques and cloud infrastructures, scalability, efficiency, and accuracy can be improved in dynamic e-commerce platforms having diverse product inventories. E-commerce firms could benefit from automating their product classification process using machine learning algorithms and cloud infrastructure as it would improve user interface, operational efficiencies, and the decision-making process.

• “Deep Learning Approach to Recyclable Products Classification: Towards Sustainable Waste Management,” by M. I. Basheer Ahmed et al. [13] - Trash management and recycling are major issues worldwide, particularly in Saudi Arabia. Manual garbage sorting is a slow process that is not efficient and prone to many mistakes. Advancements in computer vision procedures have led to waste classification automation, which enhances the efficiency of waste management. This study employs modern computer vision methodologies like CNN, DenseNet169, MobileNetV2, and ResNet50V2 to examine garbage categorization. The new findings are significant improvements over earlier works related to this problem area. The present work could help eliminate manual labor, and human errors and improve recycling techniques with the help of automating garbage classification and waste management systems for a greener future that is more sustainable than ever before.

• “Breast lesion classification using features fusion and selection of ensemble ResNet method” by Kılıçarslan G, Koç C, Özyurt F, Gül Y. Int J Imaging Syst Technol [15] - The study of breast lesion classification using an ensemble ResNet and the ALL-ResNet NCA model yielded 84.9% accuracy. In experiments, MR-MR, NCA, and Relieff performed well. Despite technological advancements, the research stresses accurate diagnosis and monitoring of breast cancer. The study utilizes ensemble classification for enhancing classification accuracy through machine learning algorithm evaluation. Majority-based voting mechanism surpasses current algorithms in breast lesion classification accuracy. Using ensemble methods and ResNet models, the study classifies breast lesions into three categories: normal, malignant, or benign.

• "Fabric defect detection and classification using modified VGG network" R. S. Sabeenian, Eldho Paul, C. Prakash [16] - A deep learning framework for AI-based fabric type and defect classification is introduced in the paper. For instance, the study has presented a modified VGG network that enhances fabric defect detection and classification to include five defects. The research reveals that deep learning can enhance quality control during the manufacturing of textiles, as illustrated by R. S. Sabeenian, Eldho Paul, C. Prakash, and Eldho Paul an assistant professor at Christ University. By being able to accurately detect and classify fabric defects the modified VGG network demonstrates how deep learning is useful in textile quality assurance.

3. METHODOLOGY

The study started with the inception of our unique dataset “dataset_v2” which was the cleaned and organized version of the raw data version “dataset_v1”, serving as the foundation element for subsequent works. Initial efforts were made in taking the traditional approach, namely in the creation of basic CNN and VGG19 architectures (Figure 3.1). However, even after several optimization attempts, their efficacy was compromised severely due to the imbalanced dataset_v2. This performance was consistent even with the TL models. This highlighted a critical need for a balanced dataset for our study. Thus the “dataset_v3” (Figure 3.1) was introduced which was the class-balanced version of “dataset_v2” with augmented and random sampled images. Although after training the models on the new and improved “dataset_v3”, there were improvements to the model’s performances but they remained suboptimal.

After this, the focus of the study was shifted to the Transfer Learning approach in which the models were imported from the Keras Application Library. EfficientNetB7 model was created in 2 phases. The first model implementation was also trained on the dataset_v2 which had the same issues as the Traditional models due to class imbalance. However, when the EfficientNetB7 was implemented in phase 2, not only was the model customized to fit better to the study’s research problem but also it was trained on the dataset_v3. This brought notable improvements to the performance of the model was observed.

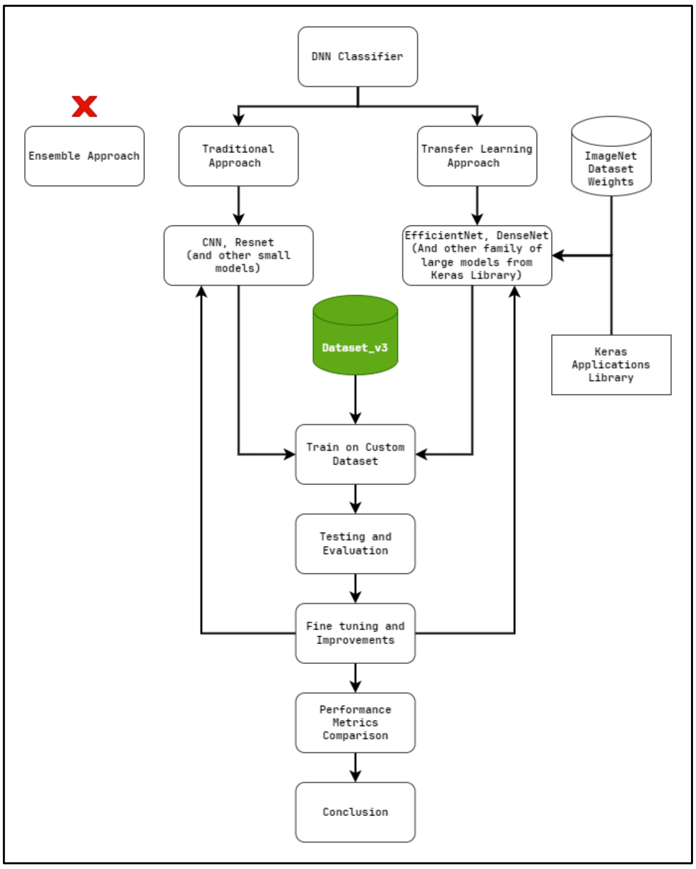

Figure 3.1: Methodology of the Study

Further exploration of Transfer Learning models leads to other complex model architectures namely, the InceptionResNetV2, DenseNet169, and DenseNet201 models. These models showcased promising classification capabilities. These models like the previous ones also underwent many adjustments and fine-tuning to enhance their performance to address the significant concern of overfitting.

3.1 MODEL CREATION AND COMPARISON

DNN classifier: The methodology of model creation focused on developing an optimal Deep Neural Network (DNN) classifier for Indian supermarket products. A comprehensive literature review established that the main DNN classifier development methods were Ensemble, Traditional, and Transfer Learning (Figure 3.1).

Traditional Approach: Despite having a strong theoretical background, ensemble techniques were not feasible due to their complexity and required expertise. Smaller traditional architectures – CNN and VGG models (Figure 3.1), were built rapidly in parallel. These models could be created easily but their effectiveness in addressing the study's problems e.g., low-quality data, limited image numbers, class imbalances, and dataset cleanliness was not good enough. Nevertheless, both conventional CNN and VGG DNNs are also developed to measure against the TL model performances.

Transfer Learning Approach: The research problem fits well with Transfer Learning (TL) technique. For faster model creation and better performance TL models' architecture was imported from the Keras Application library which had pre-trained weights from the ImageNet dataset (Figure 3.1). EfficientNet, DenseNet, and InceptionResNetV2 model architectures were picked to form the basis to enhance classification performance.

Performance Comparison and Conclusion: the study would then evaluate model performance using testing, training scores, and visual representations. The conclusions addressed the research problem and provided insights into the best model for classifying Indian supermarket products.

4. Proposed System

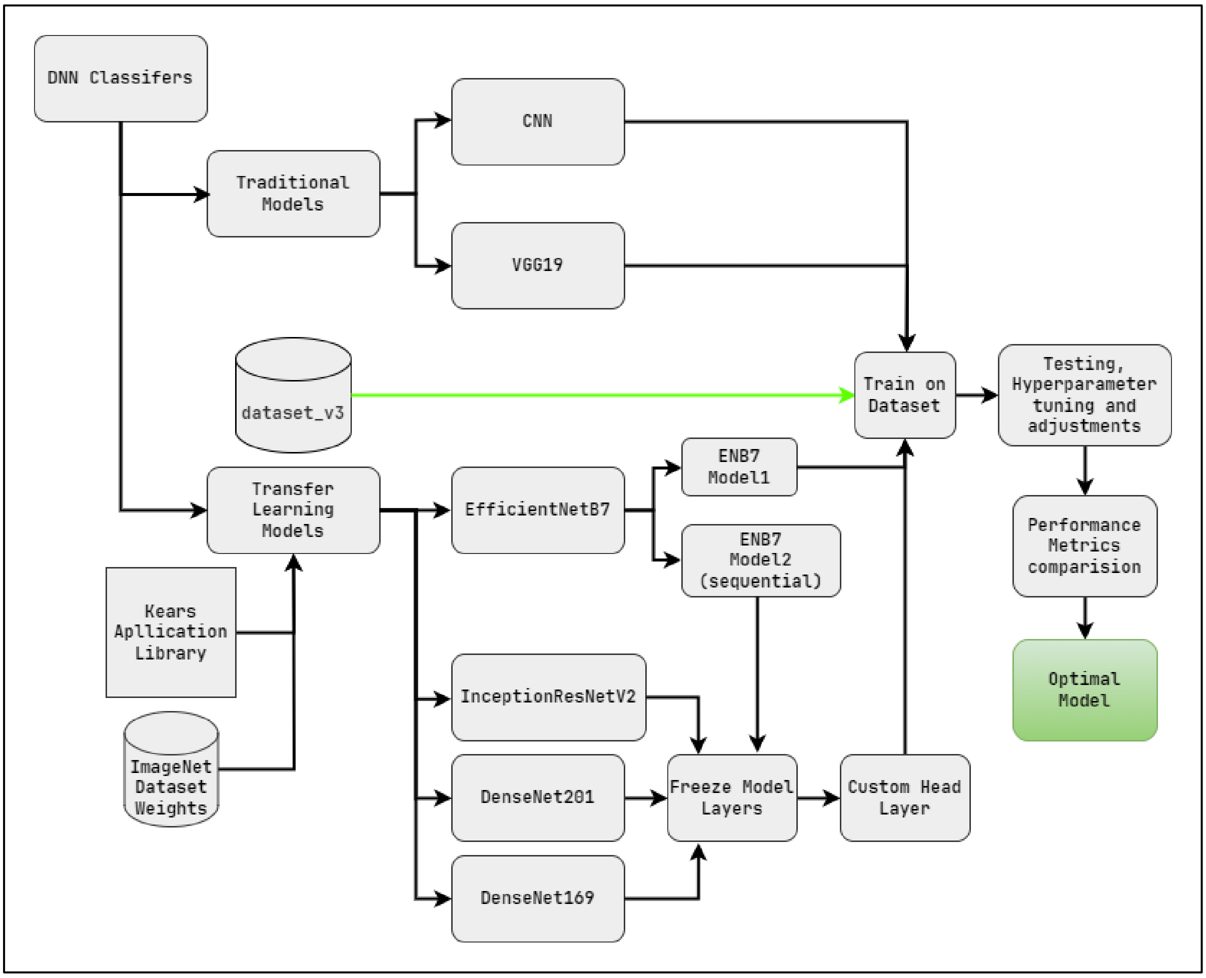

Figure 4.1: Proposed System for Model Creation and Comparison

In the proposed system for model comparison (Figure 4.6), the study adopted two distinct approaches: Traditional and Transfer Learning. First, CNN (Table 4.1) and VGG19 (Table 4.2) models were developed, layer by layer, using the Sequential method. Subsequently, both models were compiled with the Adam optimizer, categorical cross-entropy loss function, and accuracy metric for training evaluation. In initial training on dataset_v2, their performance was severely impeded by class imbalance problems. Consequently, these models were retrained on dataset_v3 while including hyperparameter tuning as well as other features like learning rate scheduler and early stopping callbacks (Table 4.1, Table 4.2). In addition to dataset_v3 producing improved performance; it was still suboptimal for product image classification, necessitating the need for a better approach to the research problem.

4.1 CNN ARCHITECTURE

Table 4.1: Architecture of Traditional CNN

BLOCK	LAYER	DESCRIPTION
Initial Block	Input Layer	The input layer specifies the shape of the input data, which in this case is a 3D tensor representing RGB images with dimensions 256x256 pixels.
Convolutional Blocks	1st Block	Consists of a Conv2D layer with 64 filters (kernels) of size 3x3 and ReLU activation function. Followed by a MaxPooling2D layer with a pool size of 2x2, which reduces the spatial dimensions of the feature maps by taking the maximum value within each pool.
	2nd Block	Similar to the first block but with an increased number of filters (128).
	3rd Block	Again, similar structure with 256 filters.
	4th Block	Similar structure with 512 filters.
Flatten Block	Flatten Layer	Flattens the output from the convolutional layers into a 1D vector to feed into the fully connected layers.
Fully Connected Block	1st Dense Layer	Consists of 1024 neurons with ReLU activation function, enabling complex feature extraction and representation
	Dropout Layer	Helps prevent overfitting by randomly dropping a fraction (50%) of neurons during training
	2nd Dense Layer (Output)	Has 15 neurons corresponding to the number of classes in the classification task (softmax activation function), producing probability distributions over the classes

4.2 VGG 19 ARCHITECTURE

Table 4.2: Architecture of Traditional VGG19

BLOCK	LAYER	DESCRIPTION
1st block	conv1_1	The first block consists of two convolutional layers with 64 filters each and is followed by a max-pooling layer.
	conv1_2	
	max-pooling	
2nd block	conv2_1	Block 2 consists of two convolutional layers with 128 filters each, and is then followed by a max-pooling layer.
	conv2_2	
	max-pooling	
3rd block	conv3_1	Block 3 consists of four convolutional layers containing 256 filters each, with a subsequent max-pooling layer.
	conv3_2	
	conv3_3	
	conv3_4	
	max-pooling	
4th block	conv4_1	Block 4 consists of four convolutional layers containing 512 filters each, with a subsequent max-pooling layer
	conv4_2	
	conv4_3	
	conv4_4	
	max-pooling	
5th block	conv5_1	Block 5 consists of four convolutional layers containing 512 filters each, with a subsequent max-pooling layer
	conv5_2	
	conv5_3	
	conv5_4	
	max-pooling	
Fattening and Fully Connected Block	FC_1	Two layers are fully connected with 4096 neurons each and are then followed by ReLU activation functions. The dense layers function as effective feature extractors, capturing advanced representations of the input data
	FC_2	
	Dropout	Dropout layers are added after each fully connected layer with a dropout rate of 0.5 to prevent overfitting. This involves randomly dropping a fraction of the neurons during training.

Final Block	Final Layer	The final output layer is composed of a dense layer containing 15 neurons, which correspond to the number of classes in the classification task (15 classes for the research requirements) When the softmax activation function is used on the output layer, it generates a probability distribution across the classes. This feature enables the model to generate probabilities for each class, facilitating multi-class classification.

In the Transfer Learning approach, we sourced models from the Keras applications library and utilized their pre-trained weights from the ImageNet dataset. The EfficientNetB7 which is known for its extensive parameterization was used in two variants:

Model 1: A simple implementation without the addition of anything apart from an extra input and output layer

Model 2: A sequential model having a custom head added, Flatten, and dropout layers. (Keras model + sequential layers)

While they seemed promising, EfficientNetB7 models showed only small improvements over classical models on both dataset_v2 and dataset_v3. Even slight improvements were only noticed when some architecture adjustments such as hyperparameter optimization, learning rate scheduler, and regularization techniques were applied.

4.3 ENB7 MODEL1 ARCHITECTURE

EfficientNetB7 Model1 uses the EfficientNetB7 architecture directly (without any changes) from the Keras applications library to create a custom model tailored for the study's particular image classification task. Given below is the architectural breakdown of this model (Table 4.3)

Table 4.3: Architecture of EfficientNetB7 Model 1

MODEL PARTS	DESCRIPTION
Base Model Initialization	The code initializes the EfficientNetB7 model from the Keras applications library with the following parameters • Excludes the top (classification) layers of the pre-trained EfficientNetB7 model, allowing for customization. • Loads pre-trained weights trained on the ImageNet dataset, providing a strong foundation for feature extraction. • Specifies the input shape of the images to be processed by the model (256x256 pixels with 3 color channels). • Applies global average pooling to the output of the final convolutional layer, reducing the spatial dimensions to a single vector.
Custom Model Construction	A sequential model is constructed using the Sequential method from the Keras library, which allows for the sequential stacking of layers. • The EfficientNetB7 base model is added as the first layer. This layer serves as a feature extractor and is set to non-trainable layers to preserve the pre-trained weights. • A flattened layer is added to convert the output of the base model into a one-dimensional vector. • A dense layer with 15 neurons and softmax activation is added as the output layer, representing the number of classes in the classification task.
Model Compilation	The model is compiled with the following parameters • Adam optimizer for gradient descent optimization. • categorical cross-entropy loss as the loss function for multi-class classification tasks. • Accuracy as the evaluation metric to monitor model performance during training.

4.4 ENB7 MODEL 2 ARCHITECTURE

ENB7 Model2 is a customized architecture derived from the EfficientNetB7 convolutional neural network (CNN) model, designed for a particular image classification task. EfficientNetB7 Version 2 functions in a manner akin to the original EfficientNetB7 model, but includes extra dense layers and dropout regularization to enhance performance and generalization.

The model utilizes pre-trained EfficientNetB7 base layers to extract features from input images, which are then processed through custom dense layers and dropout layers for transformation and regularization. Adjusting the final 20 layers (unfreezing some of the layers) enables the model to adapt to the specific classification task while leveraging the pre-trained knowledge from ImageNet. (Table 4.4)

Table 4.4: Architecture of EfficientNetB7 Model 2

MODEL PART	DESCRIPTION
Pre-trained EfficientNetB7 base model	The code loads the pre-trained EfficientNetB7 model from the Keras applications library. This model serves as the base architecture for feature extraction, initialized with weights trained on the ImageNet dataset. *Parameters:* • Excludes the top (classification) layers of the pre-trained EfficientNetB7 model, allowing for customization. • Loads pre-trained weights trained on the ImageNet dataset for feature extraction. • Specifies the input shape of the images (256x256 pixels with 3 color channels). • Applies global average pooling to the output of the final convolutional layer, reducing the spatial dimensions to a single vector.
Custom Sequential Model	A sequential model using the Keras Library's Sequential method is constructed to incorporate additional layers on top of the pre-trained EfficientNetB7 base model. *Added layers are as follows:* • A Keras Layer is added to integrate the EfficientNetB7 base model as the first layer. This layer is set to non-trainable to preserve the pre-trained weights. • A flattened layer is added to convert the output of the base model into a one-dimensional vector. • Two dense layers with 512 and 256 neurons, respectively, are added for feature transformation. These layers utilize ReLU activation functions and apply L2 regularization with a regularization strength of 0.001. • Dropout layers are inserted after each dense layer to prevent overfitting by randomly dropping 50% of the neurons during training. • Finally, a dense output layer with softmax activation is added to produce class probabilities for the classification task.
Model Compilation	*The model is compiled with the following parameters:* Adam optimizer with a custom learning rate of 0.0002. Categorical cross-entropy loss, suitable for multi-class classification tasks. Accuracy is chosen as the evaluation metric to monitor model performance during training.
Fine-Tuning	The code also unfreezes the last 20 layers of the model to allow for fine-tuning. Batch normalization layers are excluded from fine-tuning to prevent destabilizing the model.

Conversely, the InceptionResNetV2 (Table 4.5), DenseNet169 (Table 4.6), and DenseNet201 (Table 4.7) models, renowned for their complex architectures, showcased remarkable performance from the outset. These models were constructed by first freezing all of their layers and then making custom sequential models for each to make them optimized for the classification task of this research. After this, the custom head is attached to the base Keras models and compiled with the Adam optimizer, categorical cross-entropy loss function, and accuracy metric for training evaluation. Following initial training on dataset_v3, these models displayed superior performance.

4.5 INCEPTIONRESNETV2 ARCHITECTURE

Table 4.5: Architecture of InceptionResNetV2 Model

MODEL PART	DESCRIPTION
Base Model Initialization	The model code loads the InceptionResNetV2 model with pre-trained weights from the ImageNet dataset from the Keras Applications library. *Parameters:* • Excludes the fully connected layers at the top of the network, allowing for customization. • Initializes the model with pre-trained weights learned on the ImageNet dataset. • Specifies the input shape of the images (256x256 pixels with 3 color channels).
Freezing Base Model Layers	All layers of the base model are set to non-trainable to prevent their weights from being updated during training. This step ensures that only the custom head layers are trained.
Custom Head Creation	The custom head layers are defined to be added on top of the base InceptionResNetV2 model. The output of the base model is passed through a global average pooling layer to reduce the spatial dimensions of the feature maps. A dense layer with 1024 neurons and ReLU activation is added to perform feature transformation.

	Finally, a dense output layer with softmax activation is added to produce class probabilities for the classification task.
Connecting Base Model with Custom Head	The base model's input and the custom head's output are connected to create the final model. This step creates a new model architecture where the input passes through the base model's layers and then through the custom head layers.
Model Compilation	*The model is compiled with the following parameters:* Adam optimizer with a learning rate of 0.001. Categorical cross-entropy loss function, suitable for multi-class classification tasks. Accuracy is chosen as the evaluation metric to monitor model performance during training.

4.6 DENSENET169 ARCHITECTURE

Table 4.6: Architecture of DenseNet169

MODEL PART	DESCRIPTION
Base Model Initialization	The model code initializes the DenseNet169 model with pre-trained weights from the ImageNet dataset. By setting not importing the top layer, only the convolutional base of the DenseNet169 model is loaded, excluding the fully connected layers at the top. The input_shape parameter specifies the dimensions of the input images expected by the model.
Frozen Base Model Layers	After loading the pre-trained DenseNet169 model, the code freezes its layers. This prevents the weights of the base model from being updated during training, preserving the learned features from the ImageNet dataset.
Custom Head	A custom head is added to the base model to adapt it to the specific classification task for the research. The output of the base model serves as the input to the custom head. This output is stored in a variable Global average pooling is applied to aggregate spatial information across the feature maps generated by the base model. This reduces the spatial dimensions to a single vector while retaining important feature information. Subsequently, a fully connected dense layer with 1024 units and ReLU activation is added to introduce non-linearity and further abstract feature representations. Finally, a dense output layer with 15 units and softmax activation is added to produce the predicted class probabilities. The number of units in the output layer corresponds to the number of classes in the classification task.
Model Compilation	The model is compiled using the Adam optimizer with a learning rate of 0.001. Categorical cross-entropy is chosen as the loss function to measure the difference between the predicted probabilities and the true labels. The accuracy metric is specified to monitor the performance of the model during training.

4.7 DENSENET201 ARCHITECTURE

Table 4.7: Architecture of DenseNet201 Model

MODEL PART	DESCRIPTION
Base Model Initialization	The code initializes the DenseNet201 model with pre-trained weights from the ImageNet dataset by importing the model's architecture from the Keras Applications library. By not importing the top layer, only the convolutional base of the DenseNet201 model is loaded, excluding the fully connected layers at the top. The input shape parameter specifies the dimensions of the input images expected by the model (256x256 pixels with 3 channels for RGB images).
Frozen Base Model Layers	After loading the pre-trained DenseNet201 model, the code freezes its layers. This ensures that the weights of the base model are not updated during training, preserving the learned features from the ImageNet dataset.
Custom Head	A custom head is added to the base model to tailor it for the specific research classification task. The output of the base model serves as the input to the custom head, stored in a variable. Global average pooling is applied to collapse the spatial dimensions of the feature maps generated by the base model into a single vector while retaining essential feature information.

	Subsequently, a fully connected dense layer with 1024 units and ReLU activation is added to introduce non-linearity and further abstract feature representations. Finally, a dense output layer with 15 units and softmax activation is appended to produce the predicted class probabilities.
Model Compilation	The model is compiled using the Adam optimizer with a learning rate of 0.001. Categorical cross-entropy is chosen as the loss function to quantify the discrepancy between the predicted probabilities and the true labels. The accuracy metric is specified to evaluate the performance of the model during training and validation.

5. RESULTS AND DISCUSSION

This research showed the efficacy of various DNN models in the classification of Indian supermarket products on the custom dataset. The traditional model approach where custom CNN and VGG19 architectures were employed faced major difficulties associated with class imbalances within the dataset. Despite efforts to address these imbalances and fine-tune model parameters, the performance of traditional models remained suboptimal, indicating their limited suitability for the classification task for the study. The study yielded the following results:

Firstly, the creation of a new unique dataset - “Indian Grocery Image dataset_v3” consisting of more than 37K images which contained categorized, labeled images of Indian supermarket/grocery products.

Secondly, the Performance of different models varied greatly depending on the architectures and approaches used as it is seen depicted by the graphs (Figure 5.1, Figure 5.2, Figure 5.3, Figure 5.4). The accuracy and loss measures were employed to assess how effective each model was at categorizing Indian supermarket goods using given datasets. As predicted by the literature, the traditional models have difficulty when it comes to generalization while Transfer Learning models showed exceptional performance given by the increase in accuracy and decrease in loss for both testing and training scores (Table 5.1).

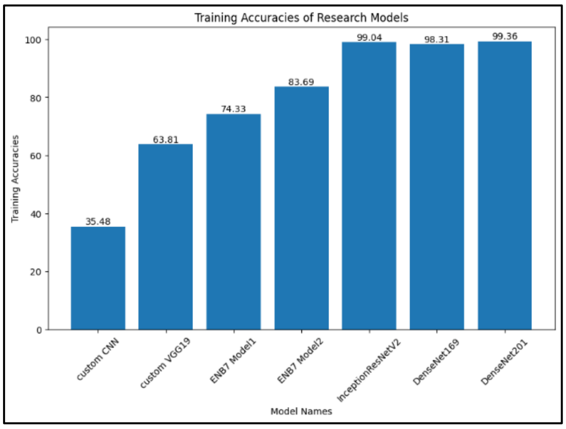

Figure 5.1: Training Accuracies of all Models

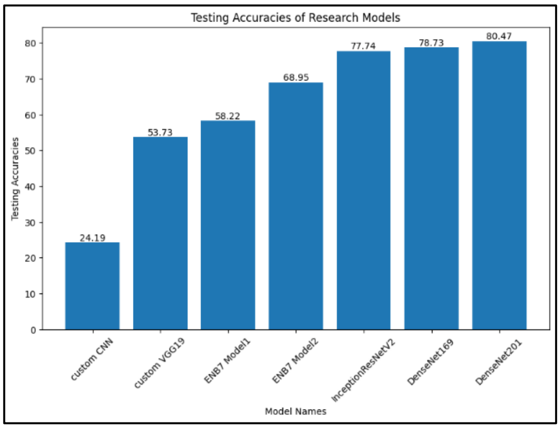

Figure 5.2: Testing Accuracies of all Models

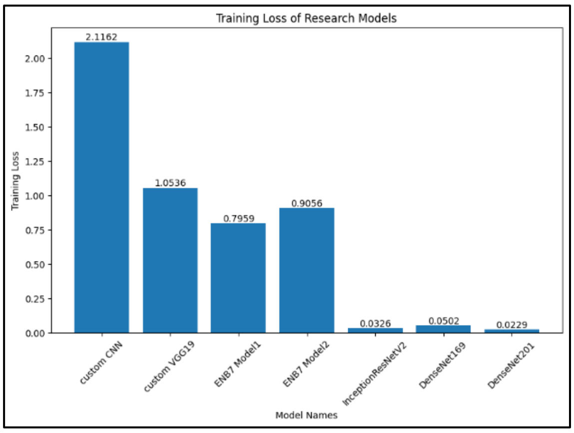

Figure 5.3: Training Loss of all Models

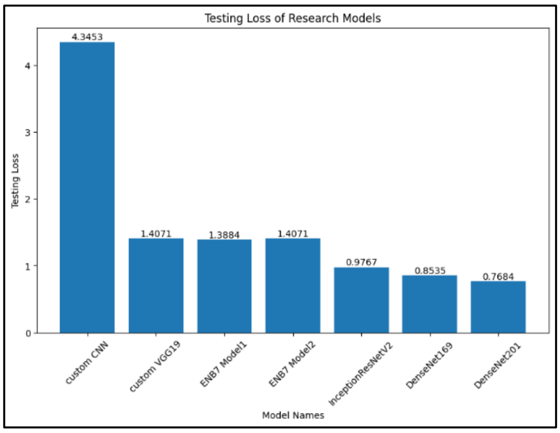

Figure 5.4: Testing Loss of all Models

Table 5.1: Comparisons of Training and Testing Scores of all Models

MODEL NAME	TRAINING ACCURACY (%)	TESTING ACCURACY (%)	TRAINING LOSS	TESTING LOSS
Custom CNN	35.48	24.19	2.1162	4.3453
VGG19	63.80	53.73	1.0536	1.4071
ENB7 Model1	74.33	58.22	0.7959	1.3884
ENB7 Model2	83.69	68.95	0.9056	1.4071
InceptionResNetV2	99.04	77.74	0.0326	0.9767
DenseNet169	98.31	78.73	0.0502	0.8535
DenseNet201	99.36	80.47	0.0229	0.7684

6. CONCLUSION

The CNN and VGG19 traditional models, although the easiest to deploy, showed relatively worse performance compared to the more complex architectures. As a result, their training and testing accuracies were quite low implying difficulty generalizing to new data. On the other hand, the EfficientNetB7 model outperformed traditional architectures. Both Model1 and Model2 of EfficientNetB7 had better training and testing accuracies where in both cases model 2 was More accurate than Model 1. Additional EfficientNetB7 models gave the lowest difference of train and test values meaning they offer the best generalization potential among all models. InceptionResNetV2, DenseNet169, and DenseNet201 remain the most accurate models as compared to other architectures implemented. These models showed good performance in terms of training accuracy as well as testing accuracy. They may effectively generalize to new data since they have relatively low values for testing loss making them optimal for this research's problem statement. Among these Transfer Learning architectures, DenseNet201 performed best relative to any other implemented model.

7. FUTURE WORKS

The study presents some areas for further investigation and enhancement. Primarily, more research is needed in data augmentation techniques and dataset balancing strategies to improve model performance, especially for classical architectures.

At the same time, it may be necessary to consider other deep neural network architecture models that were not covered in this paper, which could be found on platforms such as TensorFlow Hub and Model Zoo.

Moreover, comprehensive hyperparameter tuning experiments and ensemble learning methods are also recommended ways of optimizing the performance of the models.

Furthermore, these findings indicate the need for alternative transfer learning approaches alongside the development of domain-specific architectures and datasets for Indian supermarket product images for the practical application of such models.

REFERENCES

[1] Wiryana, I. M., Harmanto, S., Fauzi, A., Qisthi, I. B., & Utami, Z. N. (2023, September 1). Store product classification using convolutional neural network | Wiryana | IAES International Journal of Artificial Intelligence (IJ-AI). https://doi.org/10.11591/ijai.v12.i3.pp1439-1447

[2] M., Dwi Ryandra, M. F., Y. Syah, R. B., Khairina, N., & Muliono, R. (2023, July 9). Convolutional Neural Network (CNN) of Resnet-50 with Inceptionv3 Architecture in Classification on X-Ray Image. https://doi.org/10.1007/978-3-031-35314-7_20

[3] An effective CNN and Transformer complementary network for medical image segmentation. (2022, November 30). https://doi.org/10.1016/j.patcog.2022.109228

[4] Comparison of CNN-based deep learning architectures for rice diseases classification. (2023, July 14). https://doi.org/10.1016/j.aiia.2023.07.001

[5] Abdallah, S. E., Elmessery, W. M., Shams, M. Y., & A. Al-Sattary, N. S. (2022, October 8). Deep Learning Model Based on ResNet-50 for Beef Quality Classification. Retrieved from https://digitalcommons.aaru.edu.jo/isl/vol12/iss1/24

[6] Kumar, S., Pal, S., Singh, V. P., & Jaiswal, P. (2023, January 1). Performance evaluation of ResNet model for classification of tomato plant disease. https://doi.org/10.1515/em-2021-0044

[7] Anand, V., Gupta, S., Altameem, A., Nayak, S. R., Poonia, R. C., & Jilani Saudagar, A. K. (2022, July 5). An Enhanced Transfer Learning Based Classification for Diagnosis of Skin Cancer. https://doi.org/10.3390/diagnostics12071628

[8] Salehi, A. W., Khan, S., Gupta, G., Alabduallah, B. I., Almjally, A., Alsolai, H., . . . Mellit, A. (2023, March 29). A Study of CNN and Transfer Learning in Medical Imaging: Advantages, Challenges, Future Scope. https://doi.org/10.3390/su15075930

[9] Srikantamurthy, M. M., Subramanyam Rallabandi, V. P., Dudekula, D. B., Natarajan, S., & Park, J. (2023, January 30). Classification of benign and malignant subtypes of breast cancer histopathology imaging using hybrid CNN-LSTM based transfer learning - BMC Medical Imaging. https://doi.org/10.1186/s12880-023-00964-0

[10] Fusion of U-Net and CNN model for segmentation and classification of skin lesion from dermoscopy images. (2022, November 9). https://doi.org/10.1016/j.eswa.2022.119230

[11] Patra, A., Vivek, V., Shambhavi, B. R., Sindhu, K., & Balaji, S. (2021, April 16). Product Classification in E-Commerce Sites. https://doi.org/10.1007/978-981-33-4299-6_40

[12] Motwani, A., Bajaj, G., Arya, M., Sar, S. K., & Manoj, S. O. (2023, May 31). Machine Learning-Based Autonomous Framework for Product Classification Over Cloud. https://doi.org/10.1007/978-981-19-9638-2_6

[13] Basheer Ahmed, M. I., Alotaibi, R. B., Al-Qahtani, R. A., Al-Qahtani, R. S., Al-Hetela, S. S., Al-Matar, K. A., . . . Krishnasamy, G. (2023, July 17). Deep Learning Approach to Recyclable Products Classification: Towards Sustainable Waste Management. https://doi.org/10.3390/su151411138

[14] Convolutional Neural Network (CNN) Based Identification of Crop Diseases Using ResNet-50. (n.d.). Retrieved from https://ieeexplore.ieee.org/abstract/document/10370122

[15] Kılıçarslan G, Koç C, Özyurt F, Gül Y. Breast lesion classification using features fusion and selection of ensemble ResNet method. Int J Imaging Syst Technol. 2023; 33(5): 1779-1795. doi:10.1002/ima.22894

[16] R. S. Sabeenian, Eldho Paul & C. Prakash (2023) Fabric defect detection and classification using modified VGG network, The Journal of The Textile Institute, 114:7, 1032-1040, DOI: 10.1080/00405000.2022.2105112