*Grenze International Journal of Engineering and Technology, Jan Issue*

# Swimming Pools, Basketball Courts, and Badminton Courts using CNN and Transfer Learning

Valliappan Raman[1], Putra Sumari[2] and M Prabhavathy[3]
[1]Department of AI and DS, Coimbatore Institute of Technology, Coimbatore, India,
Email: valliappan@cit.edu.in
[2]School of Computer Science, University Sains Malaysia, Penang, Malaysia,
Email: putras@usm.my
[3]Department of AI and DS, Coimbatore Institute of Technology, Coimbatore, India
Email: prabhavathy.phd@gmail.com

***Abstract*— Recognizing various types of sports fields, such as swimming pools, basketball courts, and tennis courts, can be challenging due to their differing shapes, sizes, colors, and orientations worldwide. In this article, a novel approach to sports field classification is proposed using a dataset of approximately 7700 images. The classification task utilizes convolutional neural network (CNN) algorithms, which are widely employed in image recognition tasks as a deep learning technique. The results demonstrate that CNN-driven sports field classification applications can effectively automate the process of identifying different types of sports fields. This not only aids in easier identification of different types of sports field locations but also opens new avenues for sports field-related infrastructure planning and analysis. The trained model achieved an accuracy of 92.71% on the test set, validating the practicality and effectiveness of this approach.**

## I. INTRODUCTION

Sports venues such as swimming pools, basketball courts, and tennis courts form ubiquitous and integral parts of urban and suburban landscapes. The identification and classification of these facilities are paramount for urban planning, infrastructure development, and recreational activity management. To this end, this study presents a sports field detection system model using convolutional neural network (CNN) methodologies within deep learning. The system discerns different types of sports fields from images, facilitating rapid, convenient retrieval of corresponding sports venue images.

This system has applications not only in identifying sports fields like swimming pools, basketball courts, and tennis courts detected by satellites, thereby enabling individuals to input desired venues and index them on satellite maps for ease of navigation, but also in image retrieval for sports field searches. This aids individuals to quickly locate the desired sports field image on a medium such as a mobile device or a network platform, without personal retrieval, and speeds up the process of locating the desired sports field image.

Accurate and efficient sports field detection can pose a challenge due to a variety of factors. Sports fields may be viewed under diverse lighting conditions, may be obscured by other objects, and are at times difficult to visually differentiate from their surrounding environment. An ideal sports field detection system would provide accurate

*Grenze ID: 01.GIJET.10.1.326*

predictions in real-time, trained on available datasets, adaptable to different types of sports fields, and operable under different modalities (e.g., satellite imagery and drone imagery).
Sports field detection can be considered and formulated as an image segmentation problem. This study proposes a system to detect sports fields in images using CNN. This methodology constructs a convolutional neural network, compared with some models of transfer learning, eventually opting for a model with superior precision.
Therefore, the contributions of this study are:
• Developing a high-performance sports field detection system, capable of fast training using CNN with a relatively small number of images.
• Evaluating the performance of convolutional neural networks in detecting objects in images.

## II. Related Work

The realm of Convolutional Neural Networks (CNNs) has expanded significantly since their introduction by D.H. Hubel et al. [2] in 1998. Originally harnessed for image recognition, CNNs have permeated various fields, including natural language processing and drug discovery, as they demonstrate exceptional prowess in pattern detection within image data, including medical imagery [1].
Recently, the focus of CNN research has shifted towards enhancing the algorithm's performance, targeting improvements in classification outcomes. This endeavor involves integrating multiple feature extraction and classification techniques into sophisticated fusion methods [3]. Yet, the considerable time required for tuning CNN parameters poses a substantial challenge. This has catalyzed the development of swift convergence optimization algorithms, as illustrated by the MODE-CNN algorithm proposed by Ö. İnik et al. [4].
Well-established models like VGG, ResNet, and Inception are frequently leveraged in transfer learning. Additionally, models like AlexNet also offer valuable pre-training options, as elaborated by G. Limin et al. [8].

## III. Material And Methods

### *A. Dataset acquisition and pre-processing*

#### *a) Data Collection*

The data collection process involved gathering images of three types of sports fields: basketball courts, swimming pools, and tennis courts. The images were gathered from various sources, ensuring diversity in terms of location, lighting conditions, and perspectives.
The distribution of the gathered images is as follows:
*Basketball Court:* A total of 3007 images were collected. After the data cleaning process, 2,100 images were allocated to the training set, 456 images to the validation set, and 451 images to the test set.
*Swimming Pool:* A total of 2094 images were collected. After data cleaning, the remaining images were distributed into 1,473 images for the training set, 304 images for the validation set, and 317 images for the test set.
Tennis Court: Initially, a total of 2507 images were collected. Post data cleaning, 1775 images were used for training, 365 images for validation, and 367 images for testing.

#### *b) Data Preprocessing*

Once the data was collected, it underwent a preprocessing phase to enhance the efficiency and effectiveness of the deep learning model. The following are the key preprocessing techniques employed:
*Normalizing*: The pixel values of the images were normalized to fall within the range of -1 to 1 to facilitate effective learning by the CNN model.
*Type Casting*: The labels associated with each image were converted to integer data types for easy compatibility with the CNN model's output layer.
*Random Brightness Adjustment:* To improve the model's generalizability, the images' brightness levels were randomly altered within a set range.
*Random Contrast Adjustment*: Random contrast adjustments were performed on the images to enhance the model's ability to recognize sports fields under various lighting conditions.
*Random Flipping:* The images were randomly flipped horizontally to increase the diversity of the training dataset, helping the model to better generalize.
*Random Rotation*: Images were randomly rotated to a different degree. This transformation helped in enhancing the robustness of the model against orientation changes.

*Shuffling:* The training dataset was shuffled to ensure that the model does not learn any unintentional patterns from the order in which images are fed during training.
Finally, the processed datasets were inspected to verify the correctness of preprocessing steps. Sample images from the dataset were also plotted for visual verification. These preprocessing steps aided in boosting the performance of the model and its ability to generalize to unseen data.
A sample image of the dataset is shown in Figure 1:


Figure 1. Sample images of dataset

*B. Proposed CNN model*

Deep learning, particularly Convolutional Neural Networks (CNNs), is a highly effective approach for image classification. CNNs consist of convolutional layers that extract features from images by convolving filters over pixel matrices. These features are then processed through pooling layers to reduce dimensionality and fully connected layers for classification. Batch normalization and dropout techniques are often used to improve training stability and prevent overfitting. CNNs have achieved impressive results in various domains, including medical imaging, autonomous driving, and facial recognition.
This paper presents a Convolutional Neural Network (CNN) model aimed at accurately classifying images of tennis courts, basketball courts, and swimming pools. The model's design leverages deep learning technologies to automatically extract features from images and classify different types of fields. The model architecture comprises multiple convolutional layers, pooling layers, batch normalization, dropout, and fully connected layers, all designed to achieve high classification accuracy.
The feature extraction component of the neural network model consists of multiple convolutional layers, each employing 32 filters of 3x3 size and a ReLU activation function, aimed at capturing features at different scales and abstraction levels. Pooling layers are used to reduce spatial dimensions and capture the most significant features, employing a pooling window size of 2x2. To improve the stability and efficiency of the network, batch normalization is applied after each convolutional layer. To prevent overfitting, a dropout layer has been designed with a dropout rate of 0.1, aiming to retain sufficient information while reducing model complexity.
For the fully connected component, a flatten layer is used to convert the three-dimensional feature maps into a one-dimensional feature vector, which is then connected to the fully connected layers. The fully connected layer includes two dense layers with ReLU activation functions, containing 256 and 128 neurons, respectively. These layers serve as classifiers, mapping the learned features to a probability distribution corresponding to different types of fields. The final layer is the output layer, without an activation function, outputting logits values.
For model optimization, the categorical cross-entropy loss function is utilized. By comparing the model's output with the true labels, model weights are optimized.
The CNN model is shown in Figure. 2:

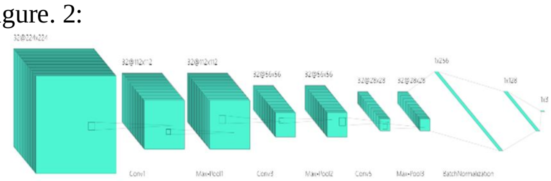

Figure 2. CNN model

*C. Classification of Swimming Pools, Basketball Courts, and Badminton Courts Using Transfer Learning*

Harnessing knowledge from one machine learning model to apply it to a new but related task is known as transfer learning. It is often referred to as 'knowledge transfer' or 'fine-tuning,' and involves the use of pre-trained models. A common technique in transfer learning is the modification of the upper layers of a static model base by appending new classifier layers and adjusting the final layers of the base model. Fine-tuning of higher-level feature representations in the base model ensures its adaptability to a specific task. VGG16 and ResNet50, transfer learning models, are employed here to benchmark their performance against a custom-developed CNN model.

*a) VGG16 Analysis*

Developed by Simonyan and Zisserman for the ILSVRC 2014 competition, VGG16 is a deep learning model composed of 16 convolutional layers, all utilizing 3x3 kernels. The design philosophy resembles that of AlexNet - that is, to progressively increase the depth of the convolution and the number of feature maps. With an aggregate of 138 million parameters, this network exhibits a highly complex architecture. The architecture in this study is tailored for the task at hand by altering the last fully connected (FC) layer, which was originally purposed for 1000 classes. The architecture is customized for the current task by modifying the last fully connected (FC) layer, which was initially designed for 1000 classes. In this study, VGG16 was reconfigured to match the classification requirements of three classes for the task, deployed with the Adam optimizer, and the accuracy metric was reported.

*b) ResNet50 Analysis*

ResNet50, an impactful deep learning model, is purposefully designed to tackle the vanishing gradient problem that frequently impedes the training of deep neural networks. ResNet50, being part of the ResNet family with an architecture of 50 layers, is recognized for its "shortcut connections" or "skip connections". These connections facilitate the flow of gradients directly through the network, simplifying the training of very deep networks.
For the task, the model of ResNet50 is adapted. This involves redefining the output layer to cater for 3 classes, as opposed to the original configuration designed for 1000 classes. Analogous to the process with the VGG models, the Adam Optimizer is utilized for training, and the model's performance is evaluated based on accuracy. The depth and efficiency of ResNet50 is leveraged to aim for a highly precise model for identifying swimming pools, basketball courts, and badminton courts.

### *D. Evaluation method*

Accuracy: The accuracy is a metric used to evaluate the performance of a classification model. It measures the percentage of correctly predicted labels compared to the total number of samples. During the training process, the accuracy values for both the training and validation sets are recorded for each epoch. The accuracy curves can be obtained by accessing model.history['accuracy'] and model.history['val_accuracy'].
Categorical Crossentropy Loss: The categorical crossentropy is a common loss function used for classification problems. It measures the dissimilarity between the model's predicted probabilities and the true labels. By minimizing the categorical crossentropy loss, the model can better fit the training data and improve the accuracy of classification. The loss function curves can be obtained by accessing model.history['loss'] and model history['val_loss'].
The formula for the categorical crossentropy loss in multi-class classification problems is as follows:

$$\text{Cross-Entropy Loss} = - \Sigma\,(y * \log(\hat{y}))$$

In this formula, the true label vector y consists of 0s and 1s, where the index corresponding to the true class label is 1 and all other indices are 0. The predicted probability vector $\hat{y}$ represents the model's output probabilities for each class.
The Cross-Entropy Loss measures the dissimilarity between the true labels and the predicted probabilities. By minimizing this loss function, the model aims to improve its ability to accurately classify samples into the correct class.

## IV. Results And Discussions

To begin with, the dataset was partitioned into training (70%), validation (15%), and testing (15%) subsets, with simultaneous execution of training and validation. Throughout the training process, this experiment examines the effect of the parameters (4.1.1), and appropriate adjustments were made to achieve improved accuracy compared to models developed using transfer learning. The deep CNN model was implemented using the "Keras" Python library, with the experiment conducted on the Pycharm platform. The experimental setup utilized the Apple M1 Pro chip, featuring specific hardware configurations of systemMemory: 16.00 GB and maxCacheSize: 5.33 GB.

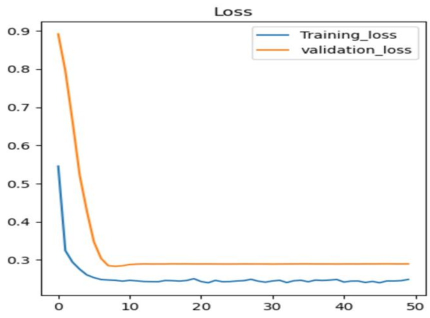


Figure 3. Cross Entrophy Loss

*A. Proposed CNN model parameters*

The model is trained using the Adam optimizer with an initial learning rate of 0.001. The learning rate is decayed exponentially using a decay rate of 0.96 and a decay step of 5. Each training batch consists of 64 samples. The model is trained for 150 epochs and employs early stopping strategy. If the accuracy on the test set does not improve for 20 consecutive epochs, the training process is terminated.

*a) Effects of hyper-parameters of the proposed model*

*(1) Effect of Batch Size*

In this experiment, the model was tested with batch sizes of 16, 32, 64, and 128. As shown in Figure 3, it can be observed that there is little change in accuracy when the batch size is increased from 16 to 32. However, there is a slight improvement in accuracy when the batch size is increased from 32 to 64. On the other hand, when the batch size is further increased to 128, there is a slight decrease in accuracy. Based on these observations, a batch size of 64 was chosen as the optimal parameter for this experiment.

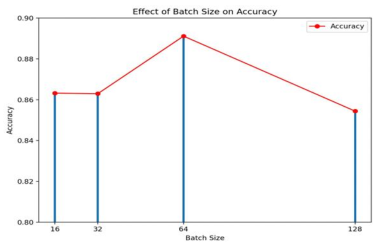


Figure 4. Effects of Batch size

*(2) Effect of Number of epochs*

The epoch refers to the number of iterations during training. In this experiment, the batch size is set to 64, the learning rate is 0.0001, and the Adam optimizer is used with an initial learning rate of 0.001. The learning rate follows an exponential decay schedule with a decay rate of 0.96 and a step size of 4. The accuracy results from 50 rounds of training are depicted in Figure 4 and 5.

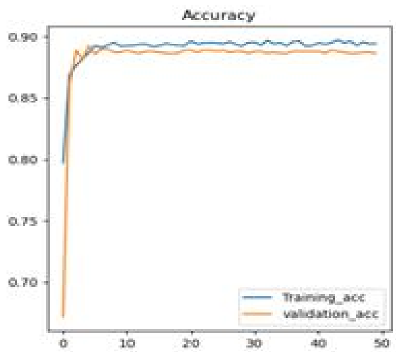


Figure 5. Accuracy of Different Epochs

The training set achieves a slightly higher accuracy compared to the test set, reaching 89.53%. Meanwhile, the test set accuracy is recorded at 88.71%. And  the loss functions of the training set and the test set show a downward trend, as shown in Figure 6.

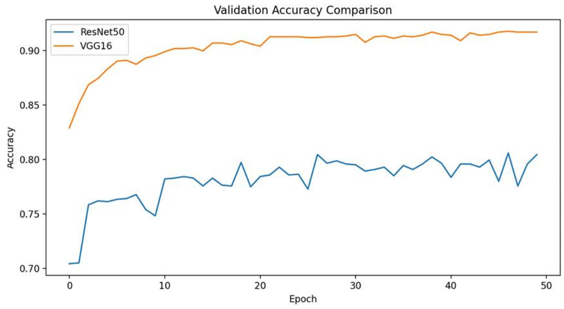


Figure 6. Loss of  Different Epochs

The results in Figure 4 and Figure 5 indicate that the model's training and learning process enables it to accurately learn the features of the training data and maintain relatively high performance on the test set.

*(3)Confusion matrix*

As Figure 7 shown, there is a confusion matrix, each row represents the actual categories, and each column represents the predicted categories. The values on the main diagonal represent the correct predictions, while the off-diagonal values represent the incorrect predictions.

These results indicate that the recognition effect of the swimming pool is the best, followed by the basketball court, and the recognition effect of the tennis court is relatively poor. The findings suggest that this study needs to adjust the model, or increase the training samples for the categories of basketball courts and tennis courts to improve the recognition performance of these categories.

*b) Comparison of classification accuracies of the proposed and transfer learning models*

The following are the optimal hyperparameter combinations The optimal hyperparameter combination obtained through hyperparameter tuning during the training process was as follows: batch size - 64, epochs - 50, optimizer - Adam, learning rate - 0.0001.

After freezing all the layers, it was found that the VGG-16 model exhibited superior performance, achieving an accuracy of 92.71% on the test dataset. In contrast, the ResNet50 model showed mediocre performance with an accuracy of only 82.2% on the test dataset. Figure 7 compares the accuracy achieved by the proposed model and the transfer learning model.

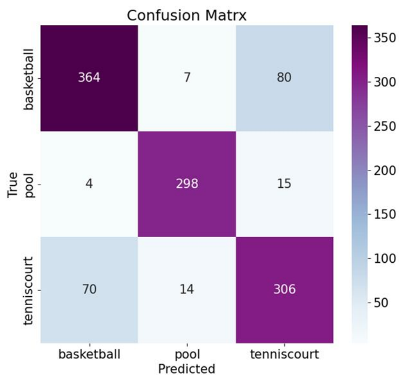


Figure 7. Comparison of Two Model Results

The results in Table 1 show the comparison of the same accuracy between the experimental model and the two migration models. The VGG-16 model shows superior performance, reaching an accuracy rate of 92.71% on the test data set. The accuracy rate of the CNN model designed in this experiment reached 88.71%, which is not bad. However, the ResNet50 model performed mediocre, with an accuracy of only 82.22%.

TABLE I. ACCURACY OF TRANSFER LEARNING MODEL AND PROPOSED MODEL

| Model | Accuracy |
|---|---|
| VGG-16 | 92.71% |
| VGG19 | 82.22% |
| **Proposed Model** | 88.71% |

The accuracy comparison on the test dataset is shown in Figure 8, which compares the performance of the CNN model constructed in this experiment with the transfer learning model.

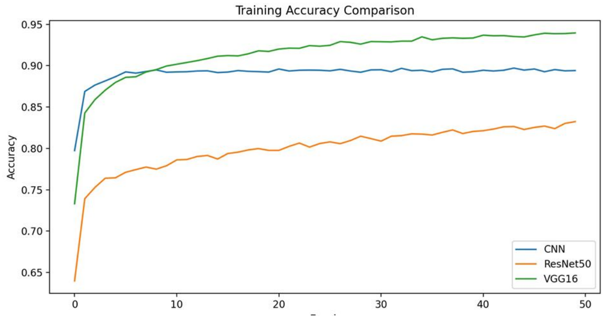


Figure 8. Comparison of Three Model Results

The results highlight the superiority of the VGG-16 model in this specific task. Its performance was better than the CNN model and ResNet50 model, demonstrating its effectiveness in capturing and extracting relevant features from image data.
The reasons for this outcome may include the following:
Firstly, the optimized hyperparameter combination chosen in this experiment may not be suitable for all models and tasks. It's possible that the hyperparameter tuning for the designed model did not achieve the best combination, while the hyperparameters of VGG-16 and ResNet50 might be more appropriate for their respective architectures. Optimizing hyperparameters is crucial for maximizing a model's performance, and further adjustments may be needed to improve the accuracy of your model.
Secondly, VGG-16 and ResNet50 are pre-trained models trained on large-scale datasets, which have learned features from a wide range of image classification tasks. Thus, transfer learning can provide a good initial feature extraction capability in many image classification tasks. Your model may not have learned sufficient general features from large-scale datasets, which could result in relatively lower accuracy.
In summary, the lower accuracy of your model compared to VGG-16 but better than ResNet50 could be attributed to inadequate hyperparameter tuning, differences in dataset characteristics, and the lack of advantage from transfer learning. These findings emphasize the importance of considering different model architectures and selecting appropriate models for specific tasks. Additionally, hyperparameter tuning and optimization play a crucial role in maximizing the performance of the chosen model.

## V. CONCLUSION

In conclusion, the study demonstrated a competent performance of a Convolutional Neural Network (CNN) based model, and transfer learning models, VGG16 and ResNet50, for the task of sports field recognition. While the model exhibited robust results, the precision for basketball courts and tennis courts was slightly less than for swimming pools.
These disparities could be attributed to potential data imbalances, where the representation of certain categories like basketball and tennis courts was considerably less than others. Another contributing factor could be the inherent feature similarity between basketball and tennis courts, which might have led to an increased rate of misclassification. Additionally, the complexity of the employed models could have resulted in underfitting or overfitting, affecting the model's performance adversely.
This research underscores the need for careful consideration of data balance, feature distinctness, and model complexity in building effective machine learning models. Future work could aim at overcoming these challenges, potentially through the collection of more high-quality data, enhancing feature extraction mechanisms, and refining model architecture.

## REFERENCES


[1] S. Kido, Y. Hirano, and N. Hashimoto, “Detection and classification of lung abnormalities by use of convolutional neural network (CNN) and regions with CNN features (R-CNN),” 2018 International Workshop on Advanced Image Technology (IWAIT), 2018. doi:10.1109/iwait.2018.8369798

[2] D. H. Hubel and T. N. Wiesel, “Receptive fields of single neurones in the Cat’s striate cortex,” The Journal of Physiology, vol. 148, no. 3, pp. 574–591, 1959. doi:10.1113/jphysiol.1959. sp006308

[3] S. P. Kannojia and G. Jaiswal, “Ensemble of hybrid CNN-ELM model for Image Classification,” 2018 5th International Conference on Signal Processing and Integrated Networks (SPIN), 2018. doi:10.1109/spin.2018.8474196

[4] Ö. İnik, M. Altıok, E. Ülker, and B. Koçer, “Mode-CNN: A fast converging multi-objective optimization algorithm for CNN-based models,” Applied Soft Computing, vol. 109, p. 107582, 2021. doi:10.1016/j.asoc.2021.107582

[5] I. Bendib, A. Gattal, and G. Marouane, “Handwritten digit recognition using DEEP CNN,” Proceedings of the 1st International Conference on Intelligent Systems and Pattern Recognition, 2020. doi:10.1145/3432867.3432896

[6] B. Muller, H. Al-Sahaf, B. Xue, and M. Zhang, “Transfer learning,” Proceedings of the Genetic and Evolutionary Computation Conference Companion, 2019. doi: 10.1145/3319619.3322072

[7] B. Tan, Y. Song, E. Zhong, and Q. Yang, “Transitive transfer learning,” Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2015. doi:10.1145/2783258.2783295

[8] G. Limin and C. Xin, “Low probability of intercept radar signal recognition based on the improved Alexnet model,” Proceedings of the 2nd International Conference on Digital Signal Processing, 2018. doi:10.1145/3193025.3193037

[9] Y. Zhou, F. Ren, S. Nishide, and X. Kang, “Facial sentiment classification based on resnet-18 model,” 2019 International Conference on Electronic Engineering and Informatics (EEI), 2019. doi:10.1109/eei48997.2019.00106

[10] M. Mehdipour Ghazi, B. Yanikoglu, and E. Aptoula, “Plant identification using deep neural networks via optimization of transfer learning parameters,” Neurocomputing, vol. 235, pp. 228–235, 2017. doi:10.1016/j.neucom