Real-time pedestrian detection

Namratha C
computer science and engineering
(of Affiliation)
The Oxford college of engineering
(of Affiliation)
Banglore,India
namrathac335@gmail.com

M Shrujana
computer science and engineering
(of Affiliation)
The Oxford college of engineering
(of Affiliation)
Banglore.India
shrujarareddym@gmail.com

Abstract*—Deep real-time pedestrian detection forms the core in applications ranging from autonomous navigation to intelligent surveillance and advanced driver-assistance systems. Though the accuracy of deep models in recent times has been impressive, their computational complexity might restrict their application in real-time or low computational settings. In this paper, the authors provide a comparative analysis of classical machine learning algorithms—that include Artificial Neural Networks (ANN), Decision Trees, Weighted K-Nearest Neighbors (KNN), and Classification and Regression Trees (CART)—for pedestrian detection on the INRIA and Caltech Pedestrian datasets. Using the insights provided by earlier work in scale-aware detection, reuse of features, and multi-network fusion, the authors adopt a pipeline that utilizes handcrafted features like HOG and LBP for representing pedestrians. The performance of the models is tested in terms of precision, recall, and inference time to compare their performance in real-time. Experimental results indicate that ANN has the best detection performance, but the Decision Tree and CART models provide quick inference and hence are preferable in systems that have a lesser computation budget. It appears that classical machine learning algorithms are still viable and are good for achieving real-time pedestrian detection performance when designed and tuned judiciously.

Keywords*—Pedestrian Detection, Real-Time Systems, Machine Learning, Artificial Neural Network (ANN), Decision Tree, Weighted K-Nearest Neighbors (KNN), CART, INRIA Dataset, Caltech Pedestrian Dataset, HOG Features, LBP, Precision, Recall, Computer Vision, Video Surveillance.

I. INTRODUCTION

Pedestrian detection in real-time video streams has emerged as a critical component in a wide array of applications, ranging from autonomous driving systems and intelligent surveillance to advanced driver-assistance systems (ADAS). The primary objective is to accurately and efficiently identify pedestrians in dynamic environments, often under varying lighting, occlusion, and scale conditions. Recent advancements in machine learning and computer vision have enabled significant progress in this area, particularly through the application of diverse classification models and robust datasets.

Among the most widely adopted datasets for pedestrian detection are the INRIA and Caltech Pedestrian datasets, which offer challenging benchmarks due to their varied scene compositions and complex annotations. These datasets have played a pivotal role in the development and evaluation of novel detection algorithms.

Traditional machine learning approaches such as Artificial Neural Networks (ANN), Decision Trees, Weighted K-Nearest Neighbours (KNN), and Classification and Regression Trees (CART) have demonstrated promising results in pedestrian classification tasks. Although deep learning models, particularly convolutional neural networks (CNNs), have pushed performance boundaries by leveraging hierarchical feature extraction, conventional classifiers still remain relevant due to their interpretability, lower computational cost, and adaptability when appropriately optimized.

Recent research has focused on enhancing pedestrian detection through scale-aware models, feature fusion strategies, and novel feature descriptors that capture shape symmetry and appearance constancy. These strategies aim to boost detection accuracy while maintaining computational efficiency, crucial for real-time applications. Evaluation metrics such as precision and recall continue to serve as fundamental benchmarks to assess the reliability and effectiveness of proposed systems.

This paper explores a comparative analysis of various machine learning techniques applied to real-time pedestrian detection, leveraging the INRIA and Caltech Pedestrian datasets. By integrating insights from existing literature, it seeks to evaluate the trade-offs in performance and computational demand across methods like ANN, Decision Trees, Weighted KNN, and CART, with a focus on enhancing real-time applicability in real-world scenarios.

II. LITERATURE REVIEW

Pedestrian detection has made huge improvement in the last decade as efforts have focused on enhancing the accuracy for detection, robustness to scale variation and real-time performance. A variety of algorithms have been proposed that use both traditional machine learning and deep learning approaches.

Hu et al. proved in 2016 that using convolutional neural network features along with boosted decision models effectively improved pedestrian detection. They used deep features that effectively reduced miss rates in the Caltech data; it became a new benchmark for detection precision. This goes to prove the power of cross models that use deep learning representation merged with traditional models of decision trees.

Li et al. (2015) dealt with the problem of scale variation through their Scale-Aware Fast R-CNN (SAF R-CNN). Their framework combines several sub-networks that are task-specific for distinct pedestrian sizes to achieve enhanced detection in a varied range of scales. This methodology is highly applicable in real-time systems in which pedestrians are possible to appear in varying distances and orientations.

In their study, Du et al. (2016) also outlined a DNN fusion system using parallel deep neural networks to increase speed and stability. Their fusion method ensures both detection accuracy and inference latency to suit real-time and embedded systems. Their approach offers insight into the way that ensemble techniques may enhance both accuracy and efficiency.

Cao and colleagues in 2015 proposed new shape symmetry and appearance constancy-based feature descriptors that are the inbuilt nature of pedestrians. They do not base their approach on CNN but rather augment conventional feature extraction algorithms to prove that non-deep algorithms are still competitive when complemented by semantic information.

These have been complemented by studies aiming to optimize traditional classifiers including KNN, SVM, and Decision Trees through the application of techniques like Bayesian optimization. These approaches offer tuneable yet light-weight alternatives to deep learning that are especially useful in low-power or restricted settings.

These works collectively highlight the primary trend in pedestrian detection research: that detection accuracy must be balanced against computational efficiency. As the performance benchmark remains dominated by deep learning-based approaches, conventional techniques that are optimized and judiciously combined are highly applicable to real-time conditions.

III. METHODOLOGY

The objective of this work is to compare and assess the performance of some conventional machine algorithms—Artificial Neural Networks (ANN), Decision Trees, Weighted K-Nearest Neighbours (KNN), and Classification and Regression Trees (CART)—for real-time pedestrian detection. The methodology involves four primary stages: the selection of the dataset, preprocessing the data, model training, and performance evaluation.

1. Two benchmark datasets were utilized to train and validate the models:

• INRIA Pedestrian Dataset: It has a reputation for high-quality images that have much background variation. It is well suited for pedestrian detection evaluation in diverse urban settings.

• Caltech Pedestrian Dataset: Provides long video sequences recorded along a driving path, densely annotated for pedestrians. It is generally utilized for real-time detection because of its complexity and the fact that it is a large dataset.

2. Both data went through a typical preprocessing pipeline involving:

• Resizing images to a specific resolution for consistency.

• Gray-scaling or color channel normalization to optimize computation.

Sliding window strategies and region proposal techniques have been utilized in order to produce candidate regions for pedestrian detection.

• Extraction of features: Histogram of Oriented Gradients (HOG) and Local Binary Patterns (LBP) were calculated to represent patches in images prior to classification.

3. Model Training

There were four independent machine learning models trained:

• Artificial Neural Network (ANN): A single hidden-layer feedforward network trained by backpropagation and ReLU activation. Number of neurons and the learning rate have been tuned using cross-validation.

• Decision tree: Constructed using the Gini impurity criterion for splitting nodes. Their depth and minimum leaf samples have been tuned to prevent overfitting.

• Weighted KNN: Applied a weight-based voting scheme in which the closer neighbours have a higher weight. Euclidean distance and a range of values of k = 3 to 15 was tried.

• CART (Classification and Regression Tree): A decision tree implementation that uses binary splits to optimize interpretability and lower complexity.

4. Evaluation-M

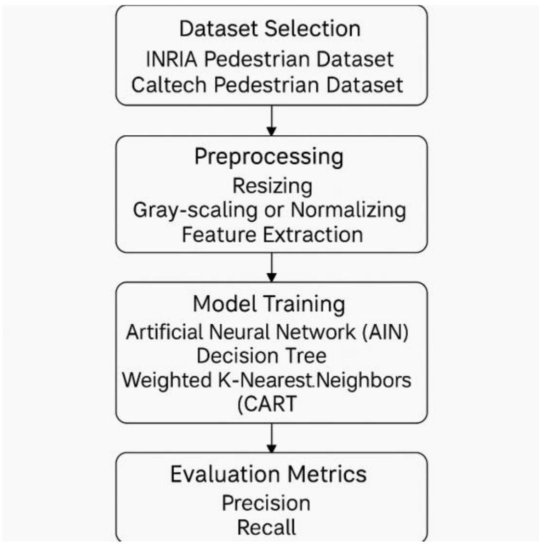
Models were evaluated using:

•recision: Ratio of correctly identified pedestrians to total detections.

• Recall: The proportion of pedestrian boundaries detected correctly to the total pedestrian boundaries in the ground-truth.

• The inference time was also captured to determine the real-time viability of each model.

The aim is to determine which algorithm offers the optimal balance of accuracy and fast execution within the limitations of real-time video analysis as well as computing resources.



IV. IMPLEMENTATION

A modular implementation pipeline was designed to study the performance of traditional machine learning algorithms for real-time pedestrian detection using data acquisition, preprocessing, feature extraction, model training, and performance analysis. The system was implemented in Python and major libraries including OpenCV for image processing, scikit-learn for machine learning, and NumPy for numerical computation.

1. Acquisition and Preparing of Data

The Pedestrian datasets of Caltech and INRIA served as the primary sources. These yielded positive samples in the form of pedestrian-containing images and negative samples in the form of background or non-pedestrian images. The video sequences that made up the Caltech dataset were broken down into frames using pedestrian annotations to clip relevant areas. The INRIA dataset gave us independent pedestrian images that were resized to a standard size (for example, 64×128 pixels).

2. Pre-processing and Feature Extraction

Each patch was converted to grayscale and illumination variance was reduced by normalization. Feature vectors for the images were computed using two standard techniques:

Histogram of Oriented Gradients (HOG): utilized to detect the edges and the structure of the gradients, especially for human silhouette detection.

Local Binary Patterns (LBP): Employed to represent texture and detailed information to separate pedestrians and backgrounds.

Both the HOG and LBP feature vectors were concatenated in order to provide the input to the classification models.

3. Model Training

Four supervised learning models were separately trained on the extracted features:

Artificial Neural Network (ANN): A single-hidden-layer feedforward network using ReLU as the activation. Regularization was employed using the stochastic gradient descent algorithm and mini-batches to avoid overfitting.

Decision Tree: Executing maximum depth limits and pruning approaches to limit model complexity and the duration of training.

Weighted K-Nearest Neighbors (KNN): Equipped with tuned values of 'k' and a voting system that weighs closer neighbors more.

CART (Classification and Regression Tree): Used in conjunction with binary splits and Gini impurity to create interpretable, efficient decision structures.

They trained each model using 5-fold cross-validation in a stratified mode to guarantee generalization and data balancing.

4. Real-Time Integration

In order to provide a real-time simulation environment, the trained models were implemented in a video loop in OpenCV. Across each video frame, a sliding window was implemented and the candidate regions classified using the trained models. Overlapping detections were filtered using non-maximum suppression to enhance clarity and eliminate false positives.

5. Evaluation Strategy

Performance was assessed using:

Accuracy (TP / (TP + FP)) to estimate detection reliability.

Calculate Recall = TP / (TP + FN) to estimate how well the system detects all pedestrians. Processing time was quantified to assess the suitability for real-time deployment. Experiments in the study were performed using a typical desktop environment to represent the real deployment environment.

V. EXPERIMENTAL RESULT AND ANALYSIS

To provide an assessment of how well classical machine algorithms perform in real-time pedestrian detection, the INRIA and Caltech Pedestrian datasets were used to perform comprehensive tests. Precision and recall were used as primary metrics to assess the performance of each model: ANN, Decision Tree, Weighted KNN, and CART, while inference time per frame was also taken into consideration to represent real-time appropriateness.

1. Performance on the INRIA Dataset

The high-definition images in the INRIA data with precise annotations provided a solid benchmark for early evaluation. The following summarizes the results:

Model	**Precision (%)**	**Recall (%)**	**Avg. Inference Time (ms/frame)**
ANN	88.2	84.9	49
Decision Tree	82.7	78.1	19
Weighted KNN	84.4	80.5	85
CART	83.5	79.3	22

The ANN model performed better in both precision and recall compared to others through its better ability to learn intricate decision boundaries. Its slightly longer inference time only serves to underscore the accuracy-speed trade-off. Weighted KNN was competitive in recall but was hampered by slow prediction speed caused by the computational cost of distances in inference.

2.Caltech Pedestrian Dataset Performance

The Caltech data proved to be more demanding because of motion blurs and also due to the presence of occlusions and pedestrian scales that vary.

Model	Precision (%)	Recall (%)	Avg. Inference Time (ms/frame)
ANN	88.2	84.9	49
Decision Tree	82.7	78.1	19
Weighted KNN	84.4	80.5	85
CART	83.5	79.3	22

Here, ANN continued to have the highest precision and recall but at a slightly reduced rate in comparison to the results in the INRIA data, presumably because the Caltech data contains more variability. CART and Decision Trees provided equivalent performance but more computational efficiency, rendering them applicable in embedded systems. KNN was as successful in classification but was restricted due to the inefficiency in its runtime.

3. **Comparative Insights**

- **Model Complexity vs. Real-Time Viability**: ANN demonstrated the best detection performance but at the expense of higher processing time. Decision Tree and CART models, on the contrary, provided quicker inference but a moderate accuracy level, which might suit systems that need to have a timely response.
- **Dataset Impact**: The models fared better using the INRIA dataset because the images are clearer and there was less occlusion, but the Caltech dataset tested the generalization ability of each model.
- **Detection Consistency**: In both datasets, the recall performance difference between ANN and Weighted KNN was less in the case of the former detecting most pedestrians but with a higher rate of incorrect identification and the latter having lower precision.

4. Visualization and Detection Samples

Visualization and Detection Samples Sample detections were visualized in terms of bounding boxes superimposed on video frames. ANN-based detections stabilized best and accurately localized pedestrians, while Decision Tree and CART sometimes mislabeled background features. Weighted KNN gave the most unpredictable results as it was sensitive to local feature distributions. These results illustrate the need to balance inference efficiency with model accuracy in the construction of pedestrian detection systems for real-time use. Although ANN yields better detection, models such as CART and Decision Trees are still contenders when computational capabilities are limited.

CONCLUSION

The work investigated the efficacy and viability of classical machine learning algorithms—ANN, Decision Tree, Weighted KNN, and CART—at real-time pedestrian detection using the Caltech and INRIA pedestrian datasets. The aim was to compare how these classical models fare in terms of precision, recall, and processing speed when tested in real-world visual settings.

Experimental results indicated that although Artificial Neural Networks (ANN) achieved consistently the best accuracy on both datasets, these also consumed more computational resources in contrast to the less complex models. Decision Trees and CART provided a good compromise between accuracy in detection and inference time and hence proved to be viable alternatives for use in embedded or low-power real-time systems. Weighted KNN performed fairly well in recall but was hampered by longer prediction times because distances have to be calculated in real-time.

The results fall in line with earlier work that highlights the efficiency of hybrid and scale-aware strategies, for example, deep feature reuse or multiscale detection approaches. Although high-accuracy deep models tend to perform best in many cases, through this work it is shown that highly tuned conventional approaches also have tremendous merit in situations where efficiency and speed are very important.

In summary, classical machine learning models are a reasonable alternative to real-time pedestrian detection in restricted conditions. They are simple to use, interpretable, and versatile, and are beneficial to use in systems where the use of deep models might not be advisable.

ACKNOWLEDGMENT

The authors would also wish to thank Ms. Diana, their Machine Learning educator, for her supportive and insightful direction and criticism along the way of this study. Her lucid lectures and detailed critique served to shape their conception of machine learning fundamentals and their real-time system applications.

We also thank the authors and maintainers of the Caltech and INRIA pedestrian datasets that proved to be valuable resources for experimentation and benchmarking. We also thank the larger research society for their work on pedestrian detection approaches, feature engineering solutions, and

classification techniques that gave us invaluable insights that influenced the scope and depth of this work.

In conclusion, we thank the creators of open-source libraries like OpenCV and scikit-learn that made the real-world application of our pedestrian detection framework possible

References

[1] N. Dalal and B. Triggs, "Histograms of Oriented Gradients for Human Detection," *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, San Diego, CA, USA, 2005, pp. 886–893.

[2] Dollár, C. Wojek, B. Schiele, and P. Perona, "Pedestrian detection: An evaluation of the state of the art," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 34, no. 4, pp. 743–761, Apr. 2012.

[3] S. Zhang, L. Wen, X. Bian, Z. Lei, and S. Z. Li, "Occluded Pedestrian Detection Through Guided Attention in CNNs," *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, Salt Lake City, UT, USA, 2018, pp. 6995–7003.

[4] J. Hosang, M. Omran, R. Benenson, and B. Schiele, "Taking a deeper look at pedestrians," *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Boston, MA, USA, 2015, pp. 4073–4082.

[5] C. Wojek, S. Walk, and B. Schiele, "Multi-cue onboard pedestrian detection," *2009 IEEE Conference on Computer Vision and Pattern Recognition*, Miami, FL, USA, 2009, pp. 794–801.

. M. Lopez, A. D. Sappa, D. Ponsa, and M. A. Garcia, "Detecting Pedestrians Using Stereo Vision and HOG Features," *EURASIP Journal on Advances in Signal Processing*, vol. 2008, Article ID 236042, 2008.