IRSTI 28.23.01 https://doi.org/10.26577/jpcsit2024-v2-i4-a4

Zh. Segizbayeva[1,*] **A. Mukysheva**[2]

[1]Economic College of Narkhoz University, Almaty, Kazakhstan
[2]Kazakh National Women's Teacher Training University, Almaty, Kazakhstan
*e-mail: segizbayeva@college-narxoz.kz

ANALYSIS OF THE EFFECTIVENESS OF OBJECT RECOGNITION METHODS IN IMAGES

Abstract. This paper considers the problem of object recognition in images, which is one of the key problems of computer vision. The relevance of the research is due to the wide application of object recognition systems in such areas as security, medicine, robotics, automotive industry and quality control. The research analyses existing recognition methods, including traditional approaches and modern deep learning methods. Their advantages, disadvantages and effectiveness in different environments are evaluated. On the basis of experimental data, the most effective algorithms for application in recognition systems were selected. The results of the work allowed us to propose recommendations for the selection and improvement of methods of object recognition, which helps to improve the accuracy and reliability of such systems. The obtained conclusions can be useful for specialists in the field of computer vision and developers of applications that use recognition technologies.

Key words: YOLOv7, object detection, Python, Google Colab, training epochs, accuracy, deep learning, computer vision.

1. Introduction

Object recognition is a task related to the identification and classification of objects in images. This task includes such steps as feature extraction, image processing and the use of machine learning algorithms for classification [1]. Traditional methods based on manual feature extraction are gradually giving way to modern approaches based on deep neural networks (Deep Learning) such as convolutional neural networks (CNNs) and transformers [2].

Despite the successes achieved, the choice of an appropriate object recognition method depends largely on the task conditions: image complexity, presence of noise, amount of training data and computational resources [3]. The main objective of this study is to comparatively analyse the performance of different methods of object recognition in images, considering the above factors

Several popular approaches, including traditional image processing algorithms, machine learning and deep neural networks [4] will be considered. The results of the analysis will identify the strengths and weaknesses of each method and provide recommendations for their application depending on the specifics of the task [5].

The classical object detection algorithm developed by Viola and Jones is based on cascade classification using Haar features [6]. The advantage of the method is its high speed due to the integral images and AdaBoost algorithm, which allows the most informative features to be extracted [7]. However, the detector is limited in application due to sensitivity to the rotation angle of objects and low accuracy in complex scenes.

The oriented gradient histogram method is widely used to detect objects such as pedestrians. It constructs histograms of gradients within localised regions of an image, which can effectively describe its structure. HOG is robust to illumination changes and easy to implement but can exhibit low accuracy when the scale and orientation of objects change [8].

2. Materials and Methods

Early developments in object recognition were based on classical image processing techniques such as: Image Segmentation, Hough Features, SIFT (Scale-Invariant Feature Transform) and SURF (Speeded-Up Robust Features) HOG (Histogram of Oriented Gradients) [9]. These methods showed good results for simple objects and conditions, they suffered from limitation in handling complex and multi-class objects and performed poorly in dealing with noise and changes in images. With the advancement of machine learning, more sophisticated and adaptive recognition methods started to emerge. Support Vector Machines (SVM) used for object classification, SVM has been extensively applied in face and car recognition tasks [10]. This method can separate data efficiently, but requires good parameter tuning and may be limited in scalability. Random Forests and Boosting These methods, such as AdaBoost and Gradient Boosting, allow improving classification accuracy by combining weak classifiers 11. Although these approaches provided higher accuracy, they still faced limitations in the context of scalability, data complexity and require significant computational resources.

With the development of deep learning, there has been a significant breakthrough in the field of object recognition through the application of convolutional neural networks (CNNs) [12]. Some of the most famous developments are AlexNet, VGGNet, ResNet, YOLO (You Only Look Once) and Faster R-CNN [13]. Real-time object detection architectures that process images quickly and can efficiently recognise multiple objects simultaneously. YOLO, for example, divides an image into a grid and predicts object classes and their locations in each cell. These methods have significantly improved object recognition results but require significant computational power, which is one of their limitations [14]. To address the limitations of deep neural networks, such as high computational load and the need for large amounts of data, hybrid methods have been developed including: Converged neural networks with pre-trained layers, Machine learning methods with data integration [5].

Current models do not cope well with low resolution or highly noisy images, requiring further development of algorithms that are robust to such conditions. Developing methods that can handle different types of objects remains an important challenge. Models trained on one type of object may have poor results when recognising objects of a different nature. Modern neural networks often act as ‘black boxes’, which makes it difficult to understand their solutions, especially in critical applications such as medicine[15,16].

2.1. The YOLOv7 method

It is a state-of-the-art algorithm for object detection that continues to build on the principles of previous versions of YOLO, improving detection accuracy and speed. Unlike older models, YOLOv7 allows efficient real-time operation, achieving high accuracy at lower computational cost. The architecture of the method includes 24 convolution layers, 4 maximum pooling (Pooling) layers and 2 full-link layers, which is illustrated in Figure 1.

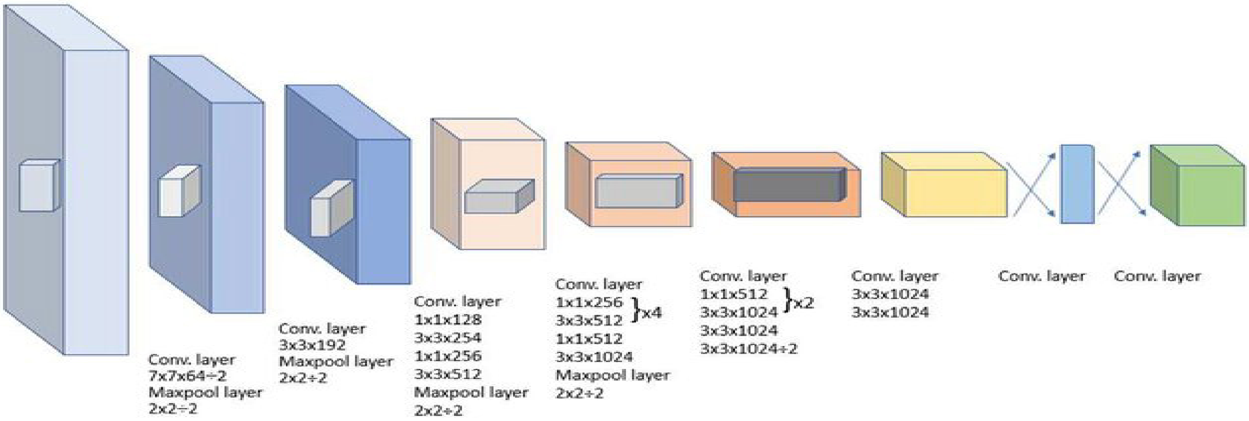


Figure 1 – YOLOv7 architecture [10]

The input image is pre-scaled to 640x640 before processing by convolution network. Initially, 1x1 convolution is applied to reduce the number of channels, after which 3x3 convolution is used to obtain the 3D output result. ReLU is used as the activation function in all layers except the last layer where linear activation is applied. Additional techniques such as batch normalisation are used to prevent overfitting of the model.

In the first step, the image (A) shown in Figure 2 is divided into a grid of size NxN with equal cells. In this case, N=4, which is shown in the right image. Each grid cell is responsible for selecting the object within it, predicting its class, and calculating the probability or confidence.

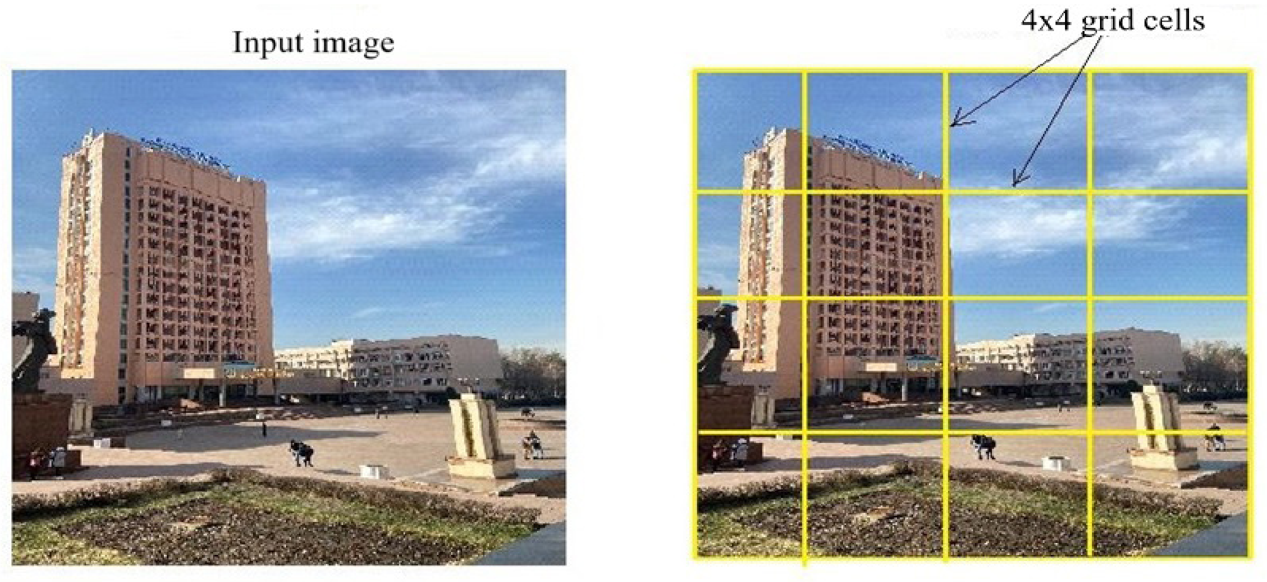


Figure 2 – Object detection using the YOLOv8 method

YOLO computes the parameters of the bounding rectangles using a single regression module that represents the following vector for each rectangle:

$$Y = [p_c, b_x, b_y, b_n, b_w, c_1, c_2] \quad (1)$$

- p_c – probability of object presence in the given bounding rectangle..
- b_x, b_y – coordinates of the centre of the rectangle relative to the current grid cell.
- b_n, b_w – height and width of the rectangle normalised with respect to the grid cell dimensions.
- c_1, c_2 – probabilities of object belonging to certain classes.

Figure 3 shows the process of calculating the parameters of the bounding rectangle using the regression approach.

In most cases, one object in an image can be predicted by several grid cells, even if their predictions do not completely coincide. To filter such cells, the Intersection Over Union (IOU) method is used, which takes values from 0 to 1. This method allows to discard unnecessary cells and leave only the most relevant ones. However, setting a threshold for IOU does not always solve the problem because an object may be associated with multiple cells whose output may create noise. To eliminate this problem, a non-local maximum suppression (NMS) method is applied, which helps to improve the detection accuracy by keeping only the best predictions.

As shown in previous tests, YOLOv7 shows significant improvement over other object detectors. It reduces the number of parameters by 40% and reduces the computational cost by 50%, while providing faster performance and real-time object detection accuracy.

The YOLOv7 architecture is based on previous versions of YOLO such as YOLOv4, Scaled YOLOv4 and YOLO-R. The main innovation in YOLOv7 is the integration of the Enhanced Effective Ensemble Level Network (E-ELAN), which improves the model's ability to learn a variety of features, promoting better learning. Figure 4 shows the depth and composite scaling implemented using the fusion-based model.

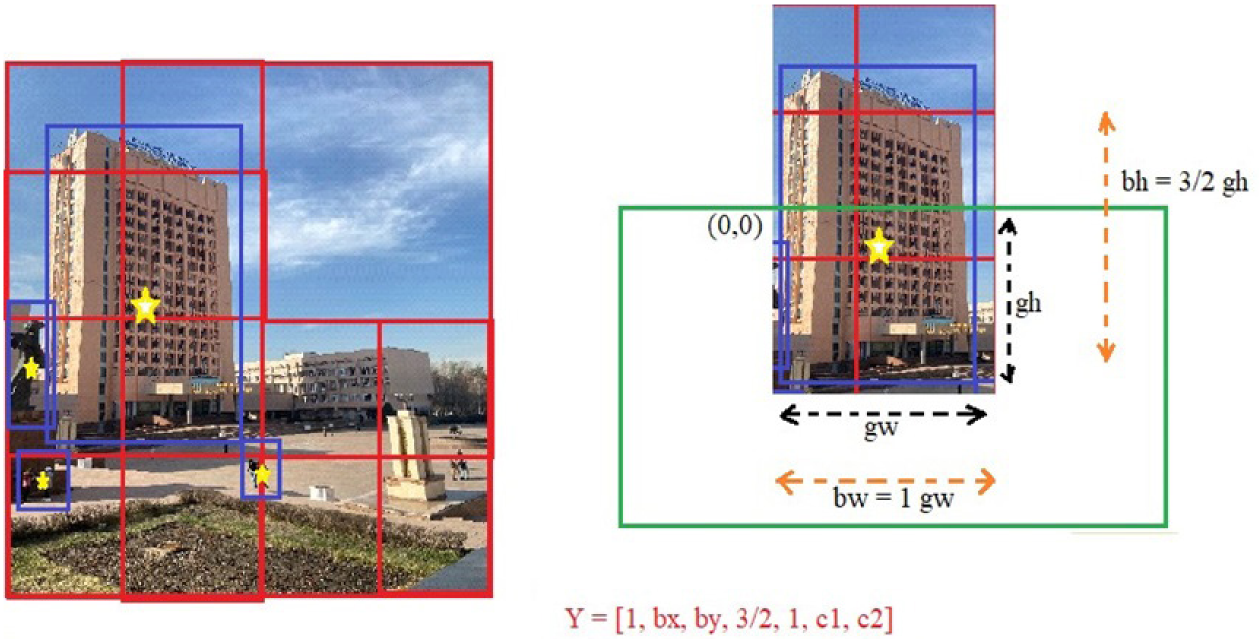


Figure 3 – Regression of the bounding box

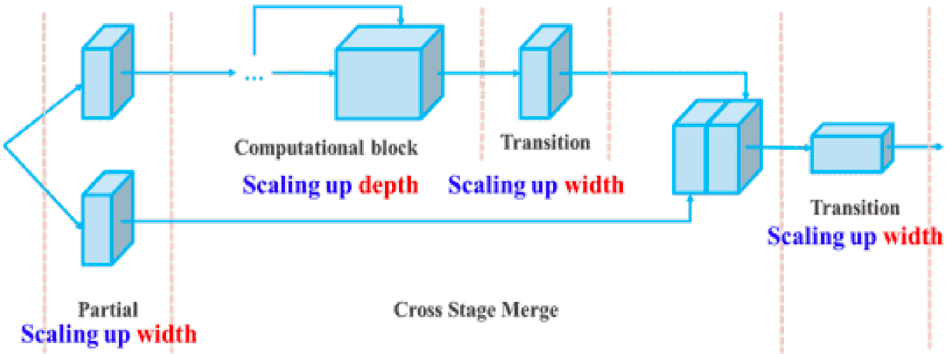


Figure 4 – Combined depth and width scaling for the fusion-based model.

2.2 The SSD method

SSD (Single Shot MultiBox Detector) [9] is based on an underlying convolutional neural network (CNN) such as VGG or ResNet, which are used to extract features from the input image. The CNN processes the image and generates a set of feature maps, each representing information about objects at different scales.

These feature maps allow SSDs to efficiently detect objects of different sizes using a multi-level architecture. Predictions are created simultaneously for multiple aspect ratios and scales, ensuring accurate and fast object detection in a single pass through the image.

SSD (Single Shot MultiBox Detector) uses feature maps extracted from multiple CNN layers to collect information about different scales. These feature maps, shown in Figure 5, have different spatial resolution and level of semantic detail.

In the example of the SSD 300 model, the input image is a frame of size 300x300 pixels. This image is first processed by the standard convolutional layers of the VGG-16 model, which extract features. Specialised convolution layers are then added to the VGG-16 output data to create feature maps for different scales. The spatial dimensionality of these maps is reduced to unity, and each of these specialised layers allows the construction of a map reflecting different image scales (see Figure 5). In these maps, each 3×33 \times 33×3 pixel region may contain a bounding rectangle corresponding to a reference designation.

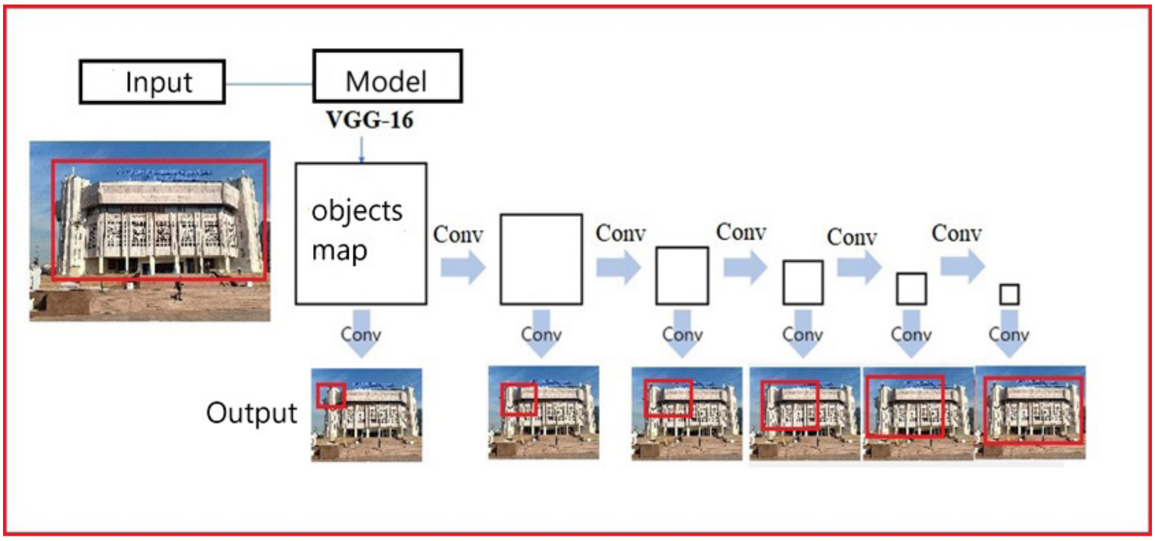
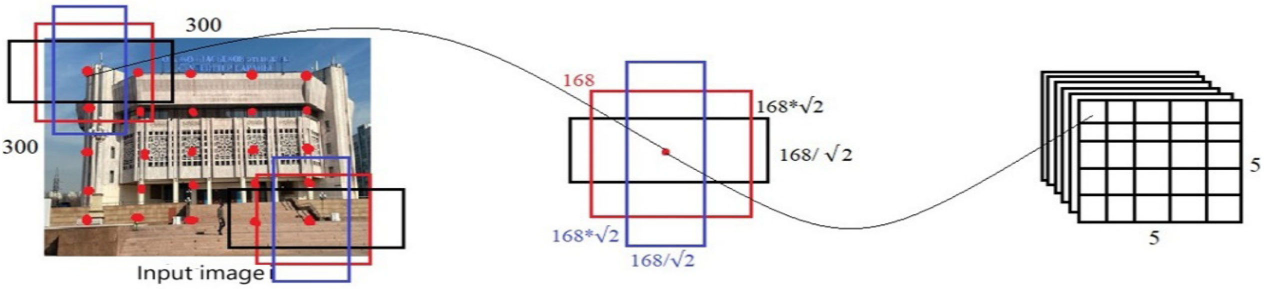


Figure 5 – Multiscale object maps for object detection using the SSD method

All feature maps are combined into a single output layer containing information about 8732 potential bounding rectangles. For each region, the following are described: feature class (C_1), centre coordinates (x,y), width (ω) and height (h) adjustment parameters of the bounding rectangle (see Fig. 6).

The non-local maximum suppression (NMS) method is applied to select a finite set of bounding rectangles from all 8732, providing accurate and efficient object detection.

SSD convolutional predictors perform predictions based on building blocks (anchor blocks) for each feature map. For each such block, SSD predicts: class probabilities for different feature categories and offsets to adjust the anchor block coordinates to match the ground truth bounding rectangles.



.

Figure 6 – Default Generetion field

Since multiple binding blocks may overlap and be associated with the same object, SSD uses a non-local maximum suppression (NMS) method. This method removes redundant predictions, leaving only the most reliable detection, blocking the others that have significant overlap (based on the IOU value).

Through the use of multi-scale feature maps and georeferencing blocks, SSD efficiently detects objects of different sizes and aspect ratios. It performs object detection and localisation in a single pass, making it faster than two-stage detectors such as Faster R-CNN. However, the weakness of SSD is the possible difficulties in detecting small objects or objects with large scale differences in a single image.

Despite this, SSD remains a widely used architecture and has served as the basis for the development of many state-of-the-art object detection techniques.

2.3. Faster R-CNN method

Figure 7 shows the flowchart of the Faster R-CNN method[8], which consists of two main components: the regional recommendation network (RPN) and the Fast R-CNN. The input image is fed through a convolutional neural network (e.g., VGG16 or ResNet), which generates a feature map. This map captures visual information at different levels of abstraction and scales. The RPN scans the feature map using small windows (anchors) at different positions and scales. For each anchor, the following are predicted: the probability of object presence (Object Estimate) and the refined coordinates of the bounding rectangle. Based on the probability, RPN filters the anchors, leaving the best NNN with the highest scores (where NNN is a hyperparameter). The filtered anchors are considered as potential regions of interest (RoI). The RoI Align procedure aligns the regions of interest to a fixed spatial size, regardless of their original size. This allows subsequent layers of the network to process RoIs efficiently. Fast R-CNN takes RoI features as input and solves two problems: assigning classes (e.g., human, car, dog) and refining the coordinates of bounding rectangles for accurate localisation. The output includes the predicted object classes and the refined coordinates of their bounding boxes. NMS eliminates redundant and overlapping predictions. The most valid predictions for each object class are retained. After NMS, final bounding boxes are left containing: object location coordinates, dimensions, Class Labels, and confidence scores.

The Faster R-CNN method provides accurate object detection and localisation. Due to its two-stage architecture, it achieves high accuracy, although its processing speed is slower than single-pass detectors (e.g. SSD or YOLO).

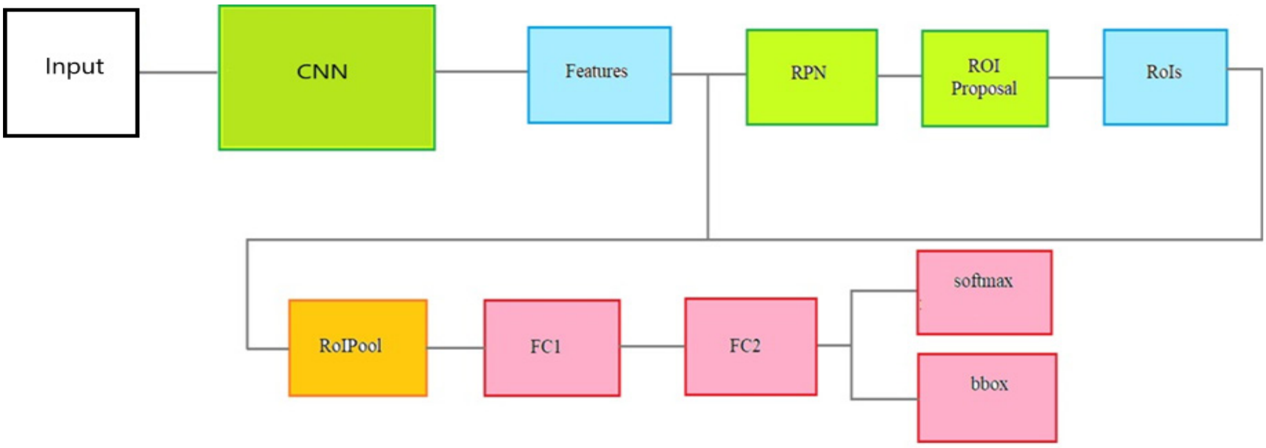


Figure 7 – Workflow of the Faster R-CNN method

Faster R-CNN offers high accuracy and efficiency in object detection by combining the Regional Recommendation Network (RPN) and Fast R-CNN. RPN generates regional suggestions, while Fast R-CNN performs object classification and bounding rectangle coordinate refinement. The joint use of convolutional features between RPN and Fast R-CNN provides end-to-end training, which allows the model to efficiently perform both tasks: sentence generation and object classification.

Table 1.1 is a comparative analysis of YOLOv7, SSD and Faster R-CNN methods, which helps to identify their strengths and weaknesses. This analysis aims at selecting the most appropriate method for a particular object recognition task.

Table 1.1 – Comparison of advantages and disadvantages of object recognition methods.

Method	Advantages	Disadvantages
YOLOv7	Very high speed of operation. - Versatility for a wide range of objects. - Compactness for embedded devices.	-Can lose accuracy when detecting small objects. - Less accuracy compared to two-stage methods.
SSD	-High speed when detecting objects of different scales. - Good balance between accuracy and performance.	- Problems with detection of small objects. - Lower accuracy compared to more sophisticated methods.
Faster R-CNN	-Faster than R-CNN due to Region Proposal Network (RPN). - High Accuracy. - Suitable for detection and segmentation.	-Complex architecture. - Performance may degrade with limited data.

3. Results

The performance of object recognition methods such as YOLOv7, Faster R-CNN and SSD is actively investigated in scientific work related to real-world object recognition and identification. These methods have been evaluated based on several key metrics: speed of performance (including FPS – frames per second). Object recognition accuracy – the ability to correctly identify and localise objects. Classification error function – a measure of errors in identifying object classes. Localisation efficiency – accuracy in determining the coordinates of bounding boxes.

Object recognition is a computer vision task aimed at identifying objects and their locations in images or video frames. The main goal is to provide accurate information about the objects present and their attributes.

Factors affecting the effectiveness of methods: The specific application (e.g. medical imaging or traffic analysis). The dataset used. The available computing resources. The requirements for speed and accuracy.

A specially created dataset was used for the experiments.:

- Objects: 7 types.
- Number of images: 1000.
- Total number of objects: 8157.
- Data split:

o 700 images for training.
o 200 images for validation.
o 100 images for testing.

Photos were taken with a mobile phone camera, which emphasizes the realism of the conditions. Each of the methods has its strengths and weaknesses. The YOLOv7 method is the leader in speed, especially useful for real-time applications. The Faster R-CNN method provides the highest accuracy due to a more complex architecture, but is slower. The SSD method balances between speed and accuracy, suitable for applications with an average load. The results of the study help to choose the appropriate method depending on the specifics of the problem and available resources.

The Python programming language and the Google Colab platform were used to train the YOLOv7 model. Training was performed on the previously described dataset, using two different numbers of epochs: 50, 100 epochs.

The results achieved at these stages of training are presented in Figures 9 and 10, which demonstrate the progress of the model accuracy depending on the number of epochs.

Training the model on the Google Colab platform allows for efficient use of GPU resources, which speeds up the training process. Achieving optimal model accuracy with different numbers of epochs illustrates the importance of choosing an appropriate number of iterations to balance accuracy and overfitting.

The Faster R-CNN, YOLOv7, and SSD methods were used in the study. Each method was extracted and trained on manually collected datasets. The error functions of the YOLOv7, Faster R-CNN, and SSD methods are shown in Figure 11.

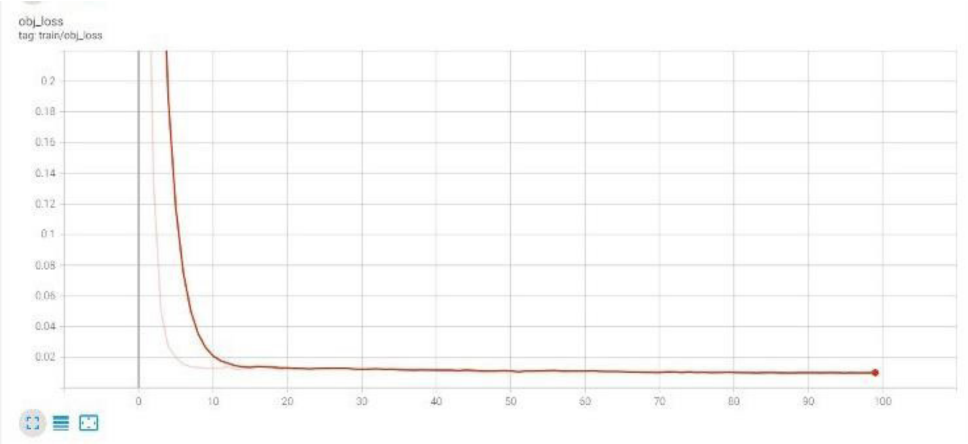


Figure 8 – Error function of the YOLOv7 method

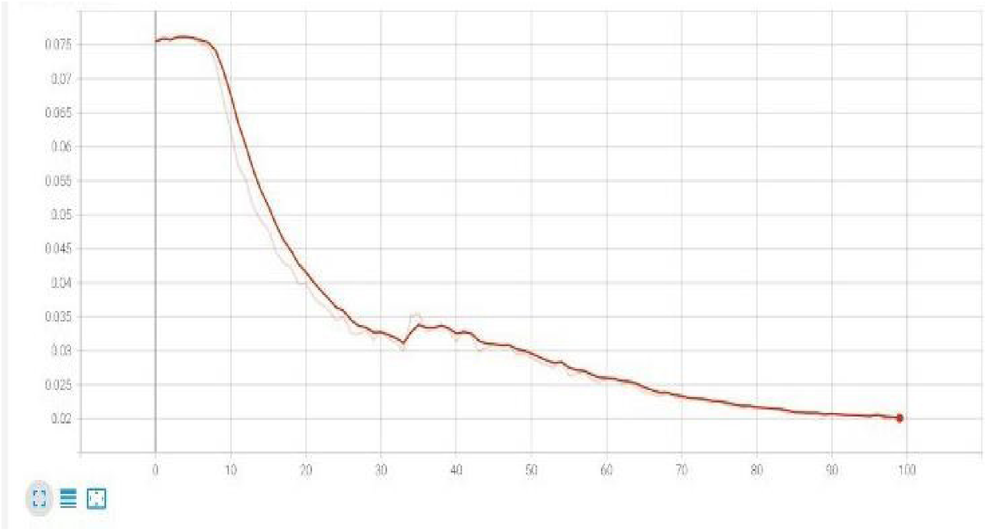


Figure 9 – Precision result of SSD method

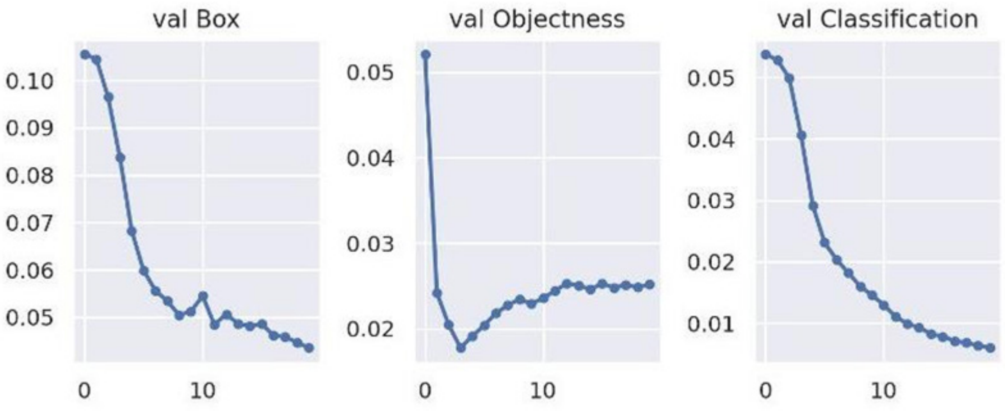


Figure 10 – Precision result of Faster R-CNN method

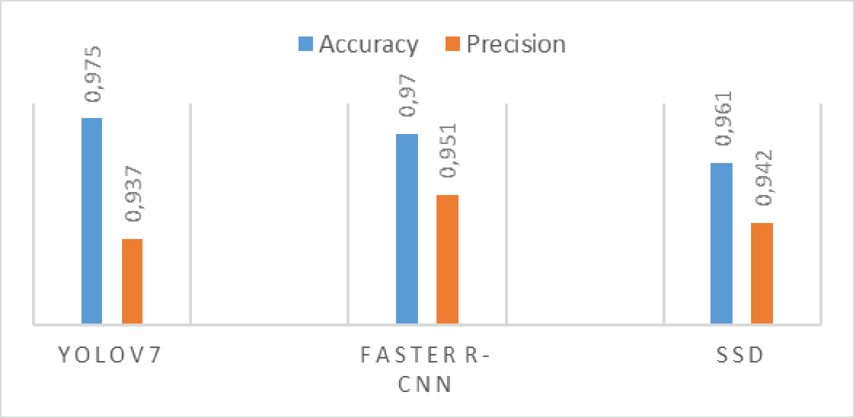


Figure 11 – Study of the effectiveness of methods.

4. **Conclusions**

In this study, three modern object detection models were analyzed and compared: YOLOv7, Faster R-CNN, and SSD. Each of these architectures has its own unique advantages and limitations, making them suitable for different computer vision tasks. YOLOv7 showed the best results in terms of detection speed and accuracy. Thanks to its single-stage architecture, the model provides high performance in real time, which is especially important for applications that require fast processing, such as video surveillance and autonomous systems. Faster R-CNN stands out for its high accuracy in detecting complex objects, especially in conditions of a large number of occlusions or small details. However, its two-stage architecture makes the model more resource-intensive, which limits its use in systems with limited computing power. SSD (Single Shot MultiBox Detector) offers a balanced approach between speed and accuracy. It is less accurate compared to YOLOv7 and Faster R-CNN, but demonstrates good performance when processing medium-resolution images and a small number of objects.

Thus, the choice of model should be based on the characteristics of the task. For real-time tasks, preference should be given to YOLOv7. Faster R-CNN is advisable to use in scientific research and applications where accuracy is more important than speed. SSD is a universal solution for systems with limited computing resources. Further research can be aimed at optimizing existing models and creating hybrid architectures that combine the advantages of high accuracy and speed.

Author Contributions

Conceptualization, Z.S. and A.M.; Methodology, Z.S.; Software, A.M.; Validation, A.M.; Formal Analysis, Z.S.; Investigation, Z.S.; Resources, Z.S.; Data Curation, A.M.; Writing – Original Draft Preparation, A.M.; Writing – Review & Editing, Z.S.; Visualization, A.M.; Supervision, Z.S.; Project Administration, Z.S.; Funding Acquisition, Z.S.

Conflicts of Interest

The authors declare no conflict of interest.

References

1. P. Viola and M. Jones, "Rapid object detection using a boosted cascade of simple features," *Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition. CVPR 2001*, Kauai, HI, USA, 2001, pp. I-I, doi: 10.1109/CVPR.2001.990517.

2. 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 vol. 1, doi: 10.1109/CVPR.2005.177.

3. A. Lohia, K. Kadam, R. Joshi, and D. A. Bongale, "Bibliometric Analysis of One-stage and Two-stage Object Detection," *Library Philosophy and Practice (e-journal)*, Feb. 2021, [Online]. Available: https://digitalcommons.unl.edu/libphilprac/4910.

4. R. Gandhi, “R-CNN, Fast R-CNN, Faster R-CNN, YOLO – Object Detection Algorithms,” Medium. Accessed: Dec. 13, 2024. [Online]. Available: https://towardsdatascience.com/r-cnn-fast-r-cnn-faster-r-cnn-yolo-object-detection-algorithms-36d53571365e.

5. S. Ren, K. He, R. Girshick, and J. Sun, “Faster R-CNN: towards real-time object detection with region proposal networks,” in *Proceedings of the 28th International Conference on Neural Information Processing Systems – Volume 1*, in NIPS’15. Cambridge, MA, USA: MIT Press, Dec. 2015, pp. 91–99.

6. K. He, G. Gkioxari, P. Dollár, and R. Girshick, “Mask R-CNN,” Jan. 24, 2018, *arXiv*: arXiv:1703.06870. doi: 10.48550/arXiv.1703.06870..

7. J. Redmon, S. Divvala, R. Girshick and A. Farhadi, “You Only Look Once: Unified, Real-Time Object Detection,” in 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 2016, pp. 779-788, doi: 10.1109/CVPR.2016.91.

8. C. -Y. Wang, A. Bochkovskiy and H. -Y. M. Liao, “YOLOv7: Trainable Bag-of-Freebies Sets New State-of-the-Art for Real-Time Object Detectors,” *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Vancouver, BC, Canada, 2023, pp. 7464-7475, doi: 10.1109/CVPR52729.2023.00721.

9. W. Liu *et al.*, “SSD: Single Shot MultiBox Detector,” in *Computer Vision – ECCV 2016*, B. Leibe, J. Matas, N. Sebe, and M. Welling, Eds., Cham: Springer International Publishing, 2016, pp. 21–37. doi: 10.1007/978-3-319-46448-0_2.

10. A. Bochkovskiy, C.-Y. Wang, and H.-Y. M. Liao, “YOLOv4: Optimal Speed and Accuracy of Object Detection,” Apr. 23, 2020, *arXiv*: arXiv:2004.10934. doi: 10.48550/arXiv.2004.10934.

11. C.-Y. Wang, A. Bochkovskiy, and H.-Y. M. Liao, “Scaled-YOLOv4: Scaling Cross Stage Partial Network,” Feb. 22, 2021, *arXiv*: arXiv:2011.08036. doi: 10.48550/arXiv.2011.08036.

12. C.-Y. Wang, I.-H. Yeh, and H.-Y. M. Liao, “You Only Learn One Representation: Unified Network for Multiple Tasks,” May 10, 2021, *arXiv*: arXiv:2105.04206. doi: 10.48550/arXiv.2105.04206.

13. K. Simonyan and A. Zisserman, “Very Deep Convolutional Networks for Large-Scale Image Recognition,” Apr. 10, 2015, *arXiv*: arXiv:1409.1556. doi: 10.48550/arXiv.1409.1556.

14. K. Kolachi, M. Khan, S. A. Sarang and A. Raza, “Fault Detection and Quality Inspection of Printed Circuit Board Using Yolo-v7 Algorithm of Deep Learning,” *2023 7th International Multi-Topic ICT Conference (IMTIC)*, Jamshoro, Pakistan, 2023, pp. 1-7, doi: 10.1109/IMTIC58887.2023.10178512.

15. G. Huang, Z. Liu, L. van der Maaten, and K. Q. Weinberger, “Densely Connected Convolutional Networks,” Jan. 28, 2018, *arXiv*: arXiv:1608.06993. doi: 10.48550/arXiv.1608.06993.

16. O. Russakovsky *et al.*, “ImageNet Large Scale Visual Recognition Challenge,” *Int J Comput Vis*, vol. 115, no. 3, pp. 211–252, Dec. 2015, doi: 10.1007/s11263-015-0816-y.

Information About Authors:

Segizbaeva Zhansaya, graduate student. Zhansaya Segizbayeva is a teacher of special subjects at the “FinTech” department of the Economic College of Narkhoz University (Almaty, Kazakhstan, segizbayeva@college-narxoz.kz). In 2015, he received a master’s degree in information systems at Taraz State University named after M.Kh. Dulaty. ORCID ID: 0009-0002-1324-9086.

Aigerim Mukysheva, She is a PhD student at Kazakh National Women’s Teacher Training University (Almaty, Kazakhstan, aigera_mukusheva@mail.ru). Her research interests mathematics, mathematical statistics. ORCID iD: 0009-0006-4384-9137.