

Image Classification and Object Detection System

Team Members:

- Nour Ahmed
- Mohamed Nagi
- Menna Mohamed
- Bola Hosny
- Noran Galal

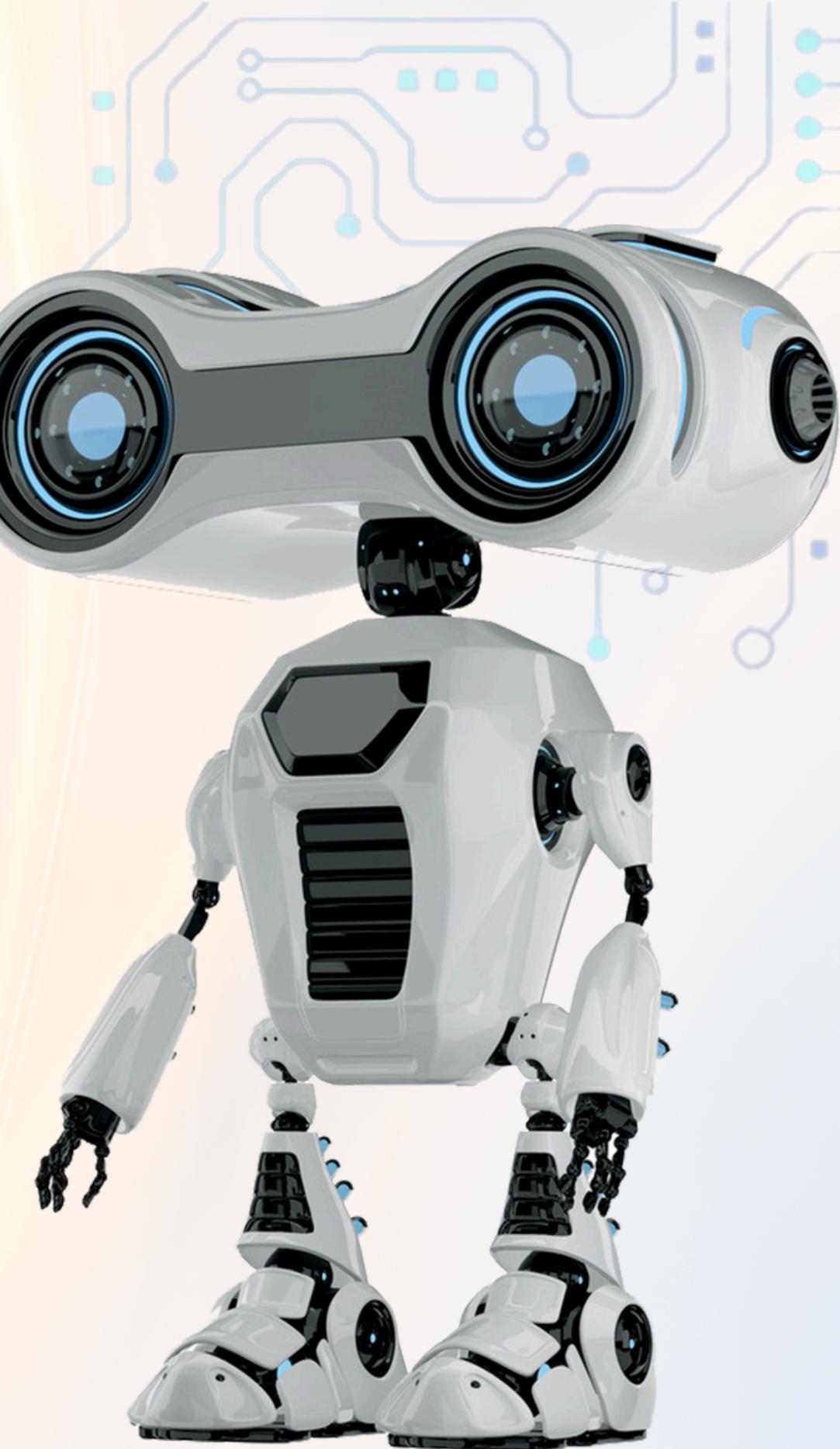
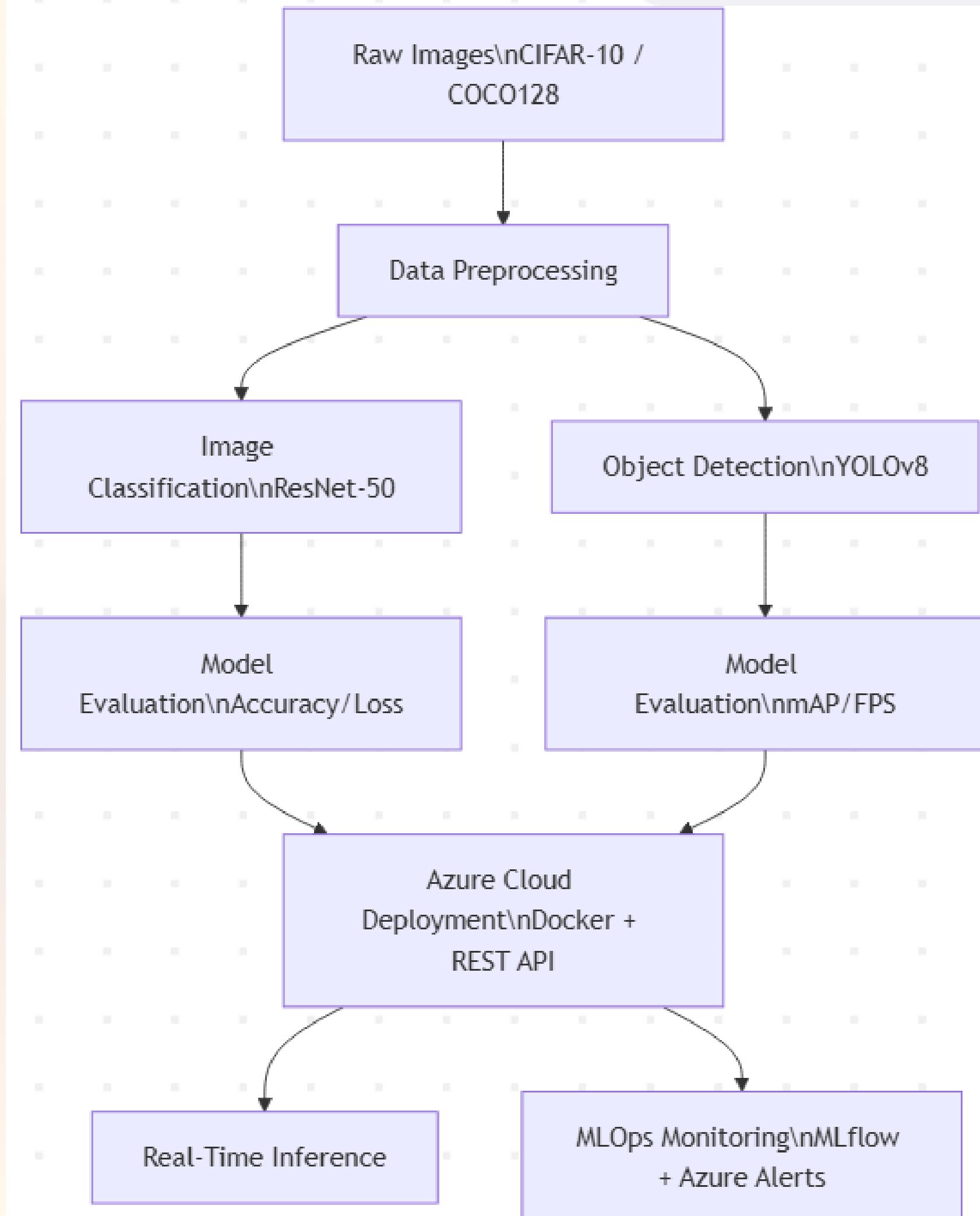


Table of Contents

1. Introduction
2. Methodology
3. Results
4. Future Improvements
5. Conclusion
6. References

Abstract

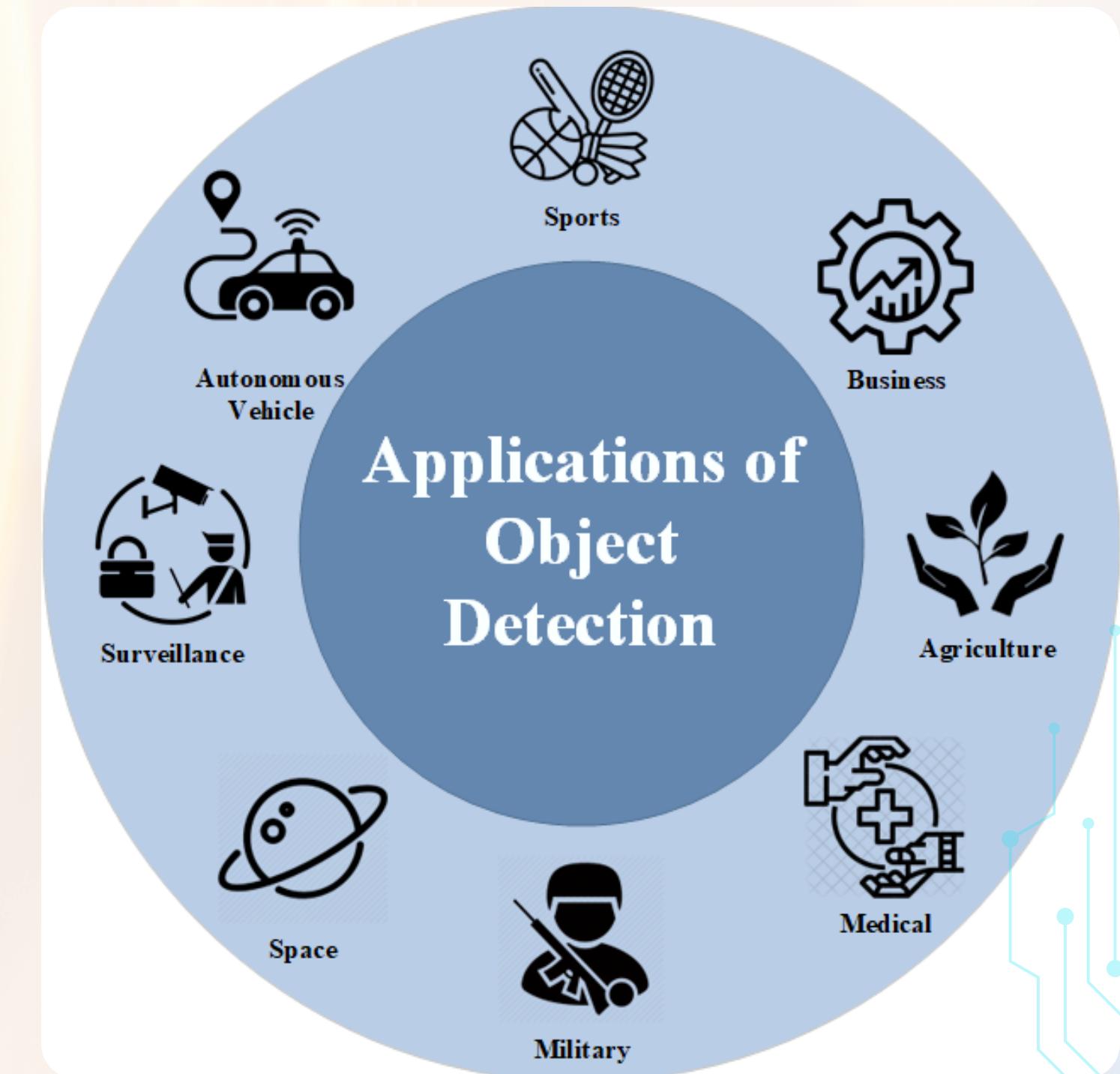
- Combines ResNet (classification) and YOLOv8 (detection).
- Datasets: CIFAR-10 (60K images) and COCO128 (128 annotated images).
- Deployed on Azure Cloud with MLOps monitoring



Why Image Recognition Matters?

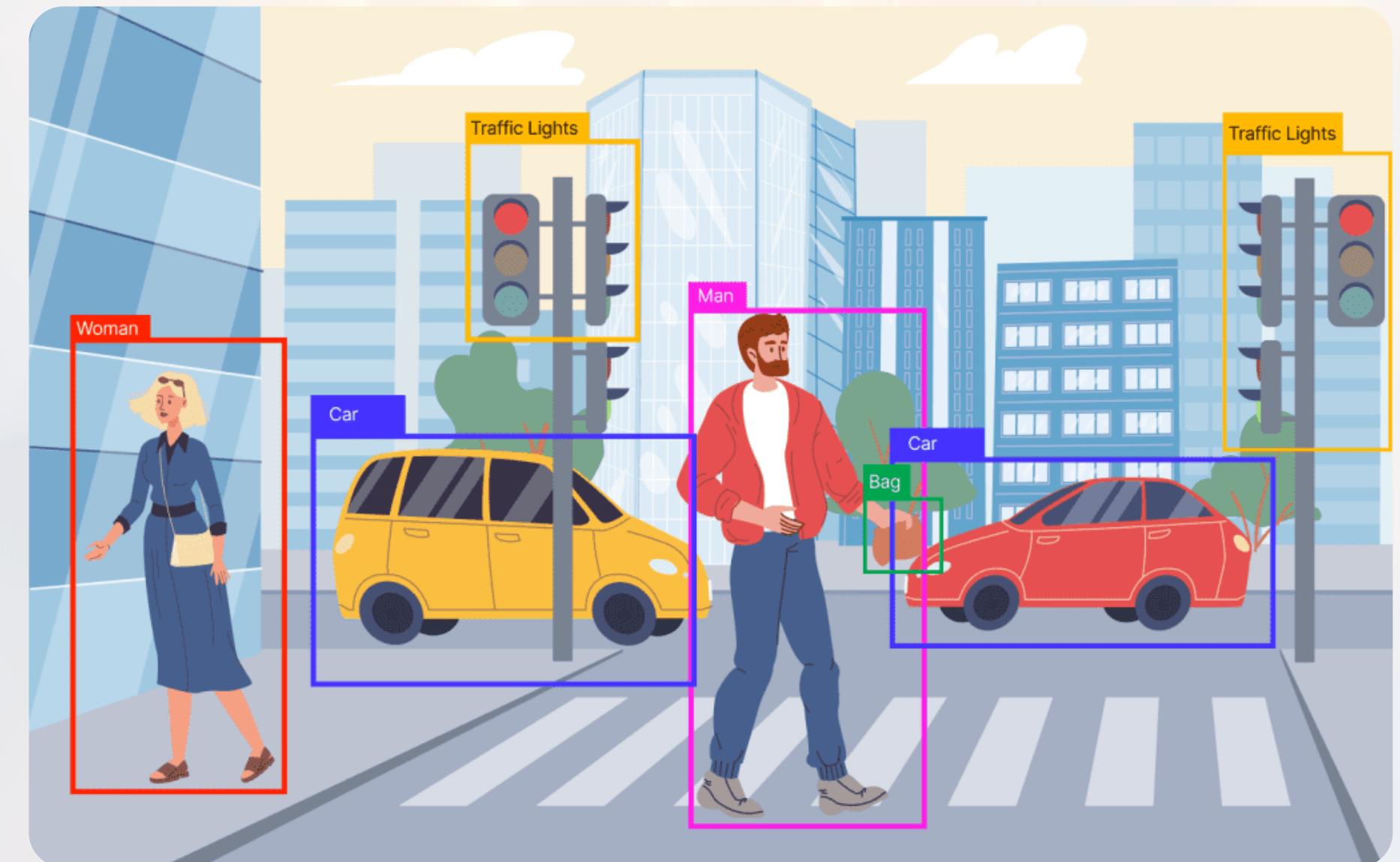
Applications:

- Autonomous Vehicles: Detection of pedestrians, traffic signs, and obstacles.
- Retail Analytics: Customer behavior tracking, inventory management, and shelf monitoring.
- Security Systems: Surveillance automation, intruder detection, and anomaly identification



Key Challenges:

- Scalability
- Real Time Processing
- Accuracy



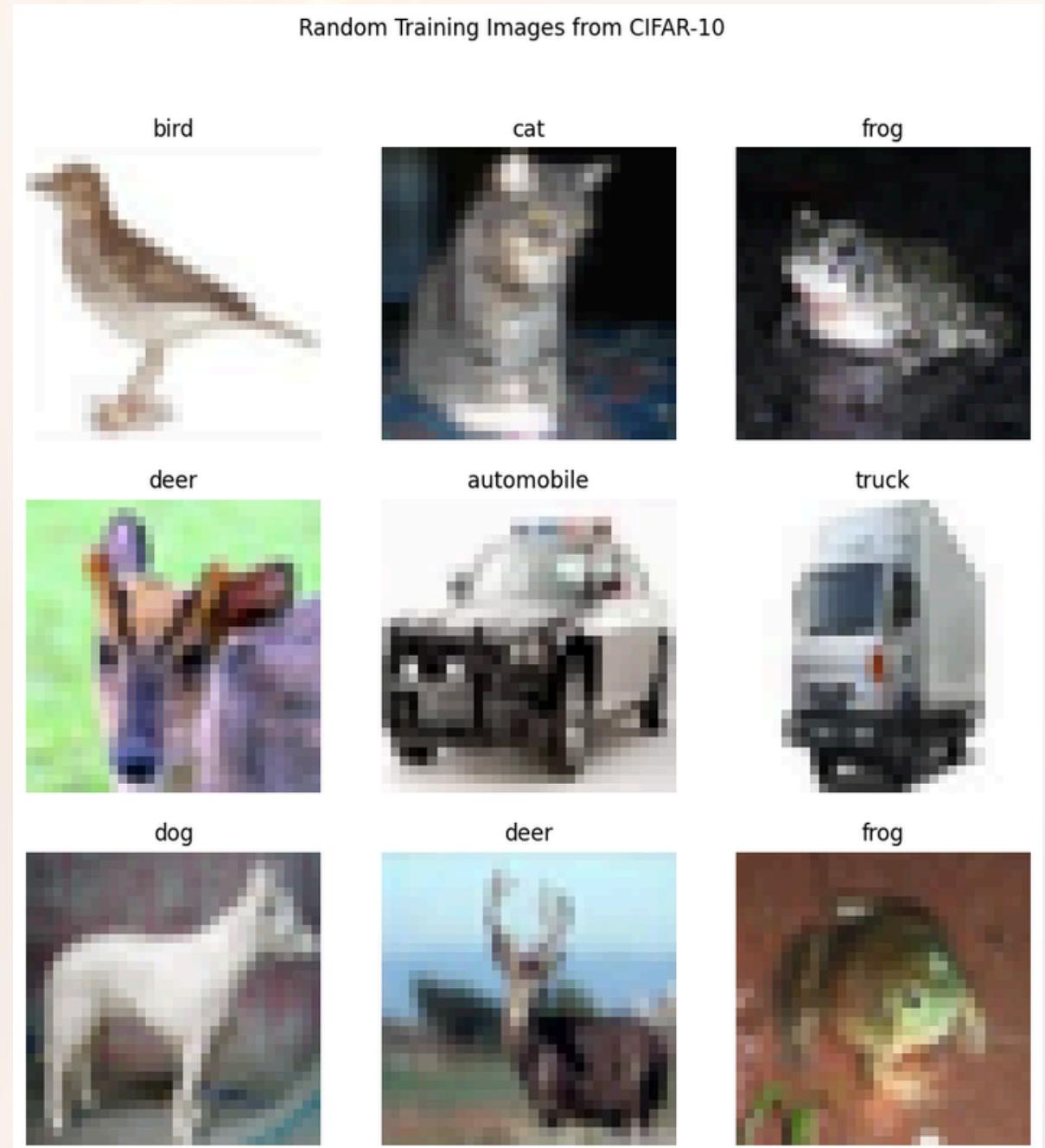
Methodology Overview

- Data Collection
- Preprocessing
- Model Development
- Deployment
- Monitoring.



Data Collection & Preprocessing

- CIFAR-10:
 - Normalized to $[0, 1]$, augmented (rotation/flips).
- COCO128:
 - Resized to 640x640, YOLOv8 internal augmentation.



Data Preprocessing

- ·CIFAR-10
 - Pixel values were normalized to the [0, 1] range to standardize input for the NN.
 - Data augmentation was applied on training images using Keras `ImageDataGenerator`, including random rotations (up to 20 degrees), width and height shifts (10%), zooming (20%), horizontal flips, and brightness adjustments.
 - Validation and test datasets were not augmented but normalized for fair evaluation
- · COCO128 Dataset
 - The COCO128 dataset, a subset of the COCO dataset with 128 annotated images across 80 object categories, was downloaded and extracted.
 - A YAML configuration file was created to specify training and validation image directories and class names for the YOLOv8 training framework.

Model Optimization

- Hyperparameter Tuning: Grid search for optimal batch size, learning rate.
- Regularization: Dropout layers (0.5 rate) to reduce overfitting.
- Transfer Learning: Fine-tuned YOLOv8 on COCO128.

Model Deployment

- Azure: Docker containers + REST API for real-time inference.
- Unified System: Single interface for classification & detection.

CONCLUSION

1. Transfer Learning

- Accelerated model development
- Achieved 82% validation accuracy (ResNet-50)

2. Cloud Integration

- Azure deployment enables scalable inference
- REST API supports real-time processing

3. End-to-End Pipeline

- Robust workflow from data to production
- MLOps monitoring ensures reliability

References

1. Krizhevsky, A. (2009). Learning Multiple Layers of Features from Tiny Images. [CIFAR-10 Dataset]. University of Toronto. Available at: <https://www.cs.toronto.edu/~kriz/cifar.html>
2. Lin, T.-Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., ... & Dollár, P. (2014). Microsoft COCO: Common Objects in Context. In European Conference on Computer Vision (ECCV). Available at: [Lorem ipsum dolor sit amet, consectetur adipiscing elit. Sed at ipsum vitae lacus lobortis lacinia. Donec tristique arcu massa, at pharetra tortor feugiat non. Etiam vehicula hendrerit aliquet. Donec sed nulla dapibus.](#)
3. He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep Residual Learning for Image Recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 770–778. DOI: 10.1109/CVPR.2016.90
4. Ultralytics. (2023). YOLOv8 Documentation. Available at: <https://docs.ultralytics.com/>
5. TensorFlow/Keras API. (2024). Keras Applications Documentation. Available at: <https://keras.io/api/applications/>
6. Azure Machine Learning. (2024). Documentation and Tutorials. Microsoft. Available at: [Lorem ipsum dolor sit amet, consectetur adipiscing elit. Sed at ipsum vitae lacus lobortis lacinia. Donec tristique arcu massa, at pharetra tortor feugiat non. Etiam vehicula hendrerit aliquet. Donec sed nulla dapibus.](#)
7. MLflow Documentation. (2024). MLflow: An Open Source Platform for the Machine Learning Lifecycle. Available at: <https://mlflow.org/docs/latest/index.html>
8. Shorten, C., & Khoshgoftaar, T. M. (2019). A survey on Image Data Augmentation for Deep Learning. Journal of Big Data, 6(1), 1–48. <https://doi.org/10.1186/s40537-019-0197-0>
9. Redmon, J., & Farhadi, A. (2018). YOLOv3: An Incremental Improvement. arXiv preprint arXiv:1804.02767. Available at: [Lorem ipsum dolor sit amet, consectetur adipiscing elit. Sed at ipsum vitae lacus lobortis lacinia. Donec tristique arcu massa, at pharetra tortor feugiat non. Etiam vehicula hendrerit aliquet. Donec sed nulla dapibus.](#)
10. Paszke, A., et al. (2019). PyTorch: An Imperative Style, High-Performance Deep Learning Library. In NeurIPS 2019. https://papers.nips.cc/paper_files/paper/2019/hash/bdbca288fee7f92f2bfa9f7012727740-Abstract.html



Thank You!