BarBeR: A Barcode Benchmarking Repository

UNIMORE
UNIVERSITÀ DEGLI STUDI DI MODENA E REGGIO EMILIA

AImage Lab
DATALOGIC

Enrico Vezzali[1], Federico Bolelli[1], Stefano Santi[2], and Costantino Grana[1]

[1]*Department of Engineering "Enzo Ferrari", University of Modena and Reggio Emilia, Italy*
{name.surname}@unimore.it
[2]*Datalogic, S.p.A, Bologna, Italy*
stefano.santi@datalogic.com

1 – Introduction

BarBeR

Barcodes have been essential for automatic data capture for over seven decades, playing a critical role in industries such as retail, manufacturing, and logistics.

Research in this field is hindered by a lack of public datasets and available code implementations.

*To tackle these challenges, we introduce **BarBeR—an open-source benchmark** for barcode detection paired with a **public dataset of 8,748 annotated barcode images**.*

Link to Repository *Link to Dataset*

2 – Dataset

Barcode types: 19 classes, including linear (Code 128, UPC) and 2D (QR Code, DataMatrix).

The dataset includes images captured under various conditions, such as varying lighting, noise, and obstructions.

Annotations: VGG format with polygon shapes, barcode type, pixels-per-module (PPM), and encoded strings.

3 – Benchmark Framework

BarBeR supports a plethora of algorithms that could be tested with different tests and metrics:

- **Available Algorithms**: supports four traditional CV methods (**Gallo *et al.*, Soros *et al.*, Yun *et al.*, Zamberletti *et al.***) and three deep learning frameworks (**Torchvision, Ultralytics** and **Detectron2**).
- **Available Tests**: single-class **localization** (1D or 2D), multi-class **detection**, **time measurement.**
- **Available Metrics:** Precision, Recall, F1 Score, mAP@IoU.

4 – 1D Barcode Localization

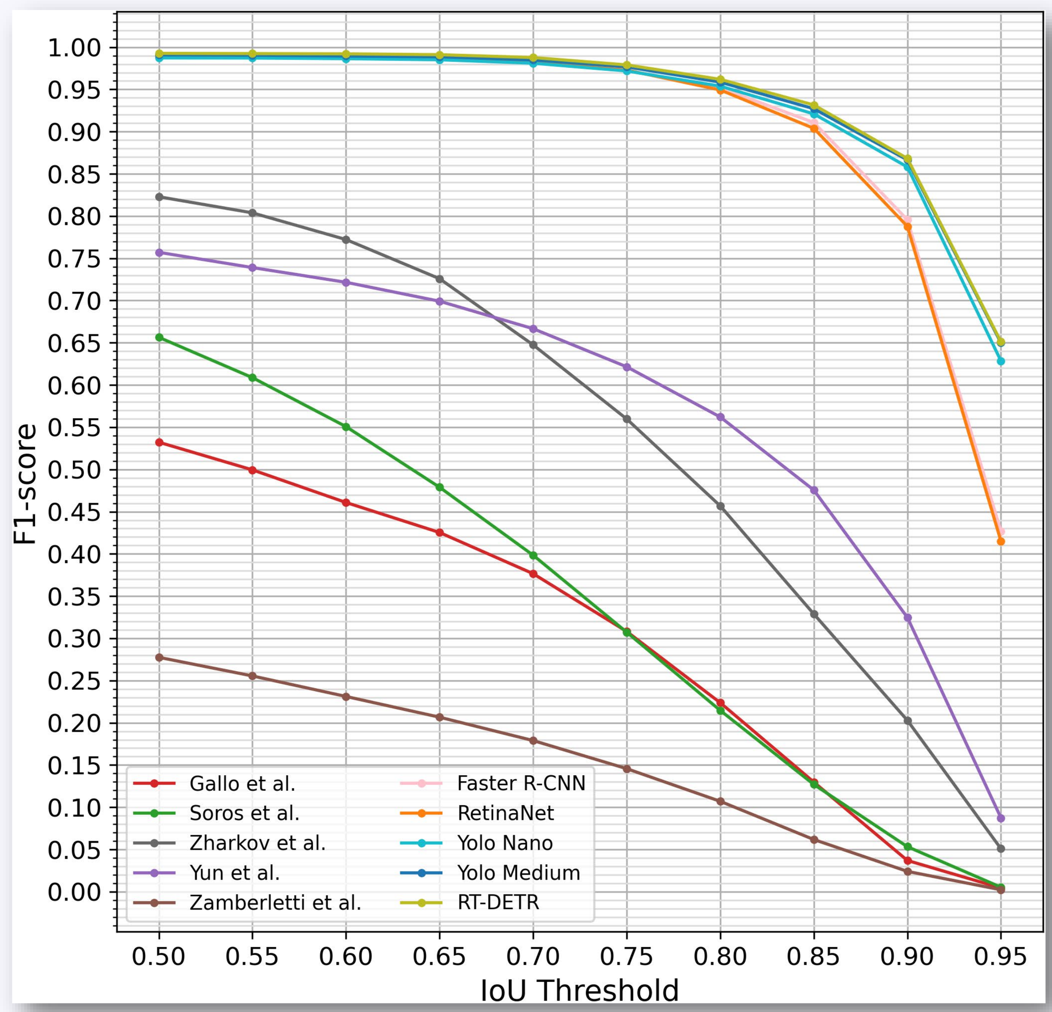
The framework allows for **graph visualization of different metrics**.

All the available algorithms support 1D barcode localization.

The test was performed with 5-fold cross-validation

Deep-learning methods have higher F1 scores, with RT-DETR achieving the highest score.

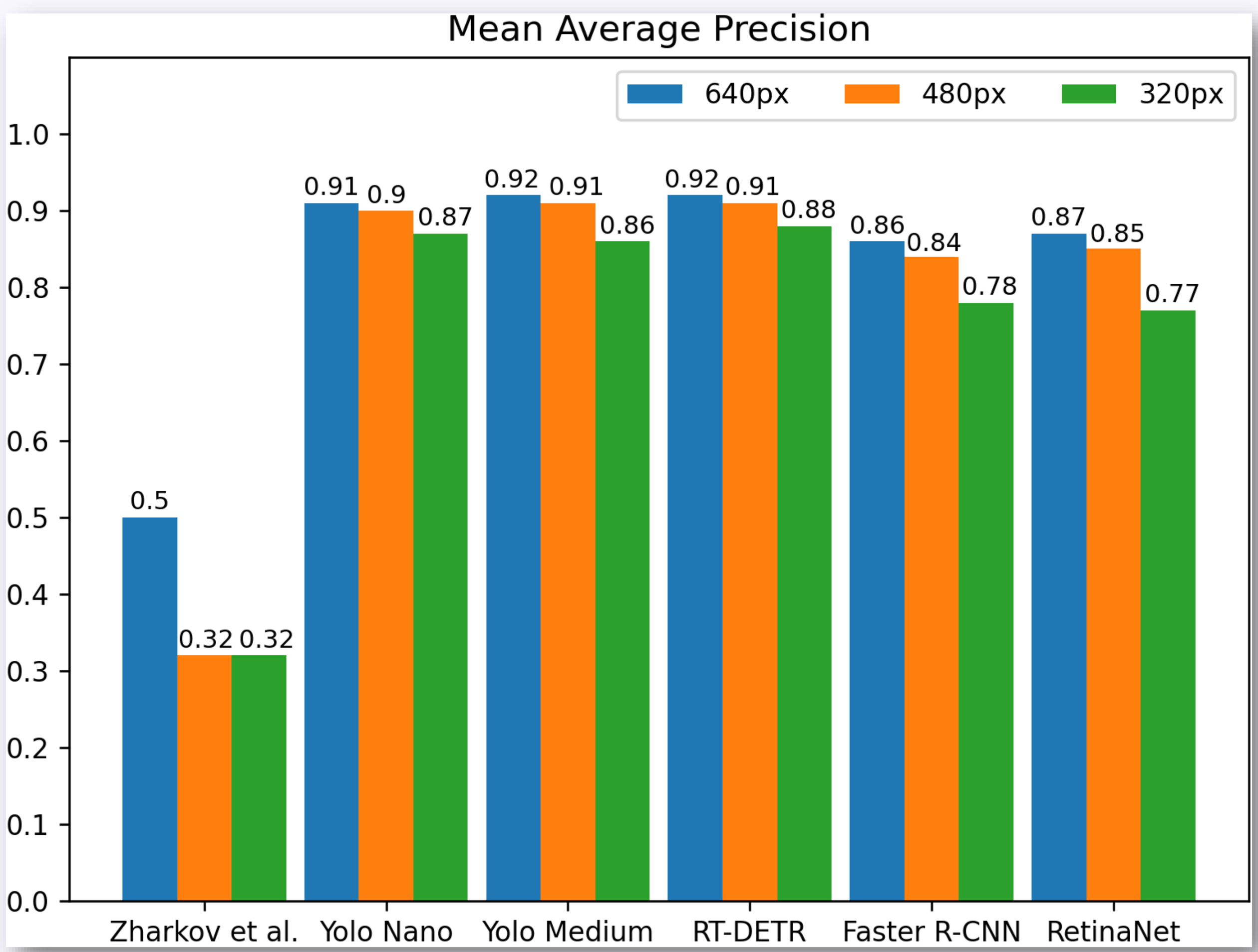
Yun et al. is the best among traditional CV methods.



F1-scores for 1D Barcode Localization.

5 – Multi-class Detection

Multi-class detection is supported only by deep-learning models. Models are tested at different image resolutions.



mAP@[0.5:0.95] Multiclass Scores.

6 - Time measurement

Algorithms were tested on a high-end PC and a Raspberry PI 3B+.

This highlights significant differences in processing times based on the hardware and method used.

Detection Method	Times on PC (ms)			Times on Raspberry PI (ms)	
	Single-Thread CPU ↓	Multi-Thread CPU ↓	GPU ↓	Single-Thread CPU ↓	Multi-Thread CPU ↓
Gallo *et al.*	**1.63**	-	-	**53.45**	-
Soros *et al.*	11.25	-	-	397.53	-
Zamberletti *et al.*	48.20	-	-	1 360.23	-
Yun *et al.*	7.59	-	-	146.31	-
Zharkov *et al.*	25.85	**5.97**	**1.45**	2 120.43	1 949.08
YOLO Nano	64.99	17.40	18.66	3 034.27	1 803.09
YOLO Medium	478.92	51.36	23.91	20 083.87	15 813.46
RT-DETR	985.41	141.06	37.55	39 882.45	33 224.15
Faster R-CNN	1 271.93	237.91	30.27	∞	∞
RetinaNet	1 124.11	105.20	36.00	∞	∞

Conclusions: most methods can run in real-time on PCs, but mainstream deep learning architectures remain too slow for many embedded applications.