Normalizing Flow based Feature Synthesis for Outlier-Aware Object Detection

Nishant Kumar[1], Siniša Šegvić[2], Abouzar Eslami[3], Stefan Gumhold[1]

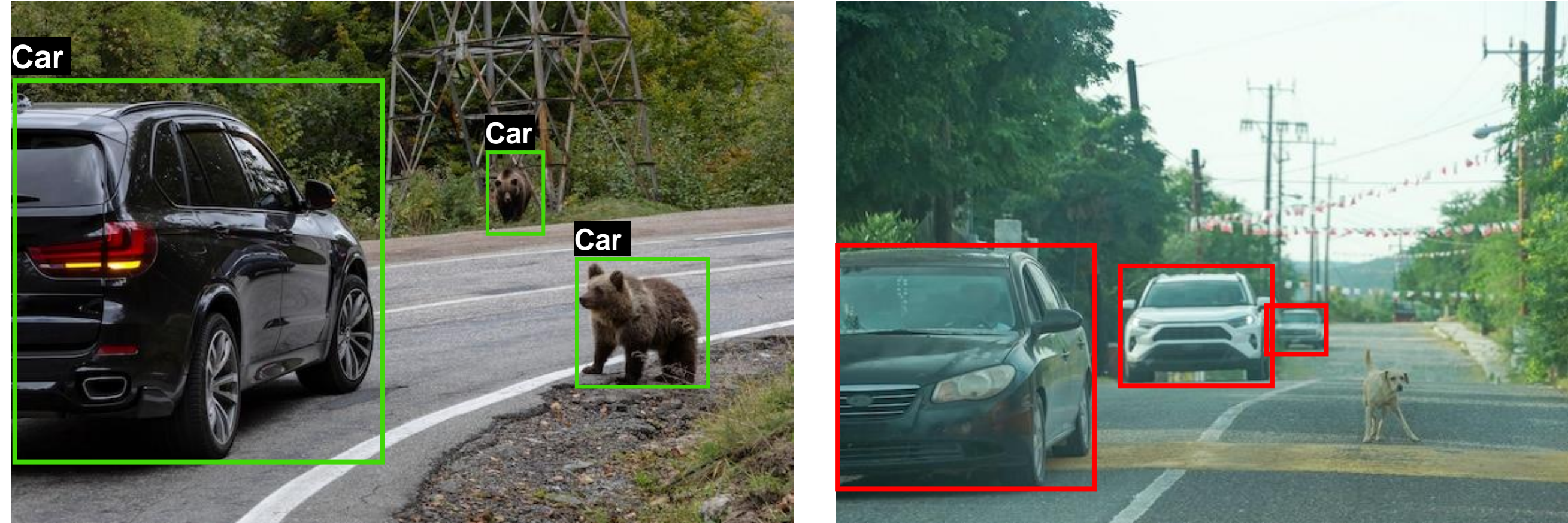[1]TU Dresden, [2]University of Zagreb – FER, [3]Carl Zeiss Meditec AG

Motivation

➤ Vanilla Object Detectors like Faster R-CNN are not reliable to distinguish outliers from inliers.
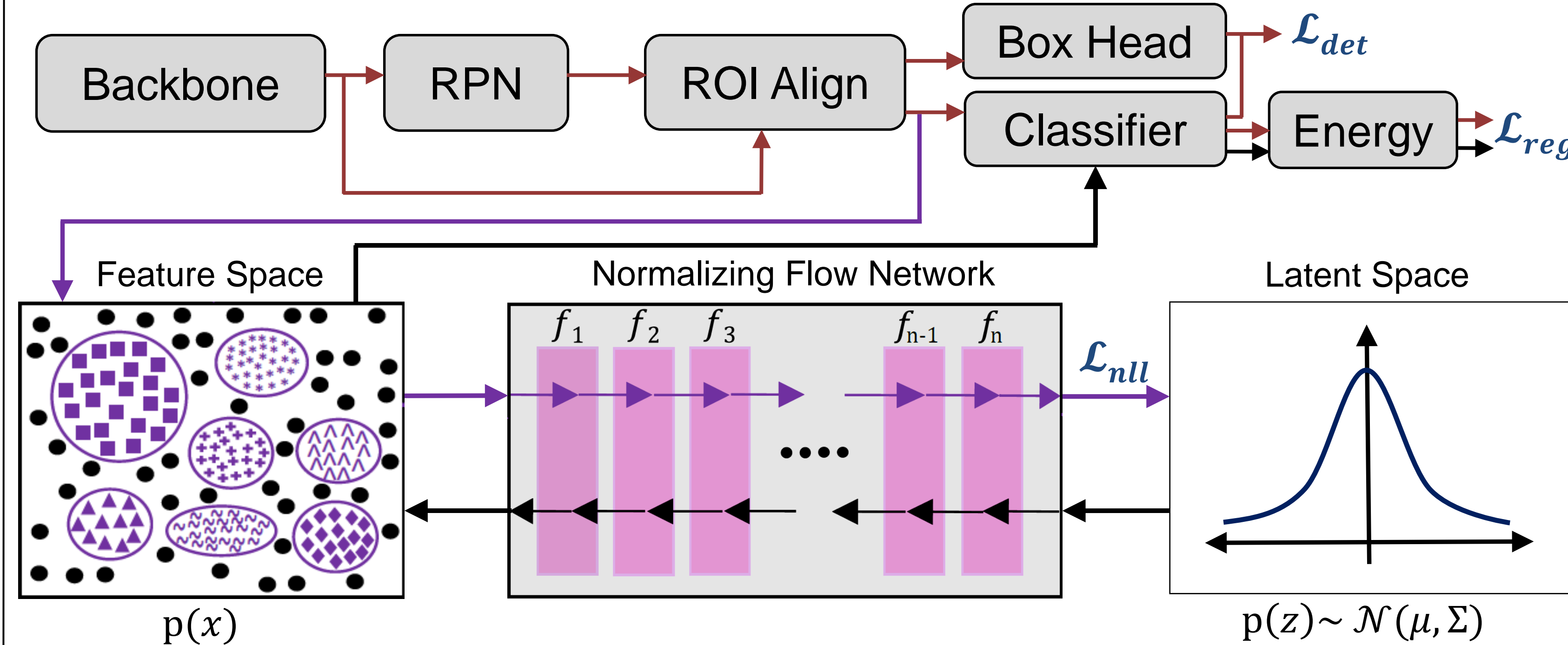

➤ Prior works synthesize outliers using class-conditional Gaussians for training an outlier-aware object detector.

➤ A synthesized outlier may have higher likelihood with respect to another inlier class when compared with its parent class.

Inliers - Class A

Inliers - Class B

Synthesized Outlier from Class A
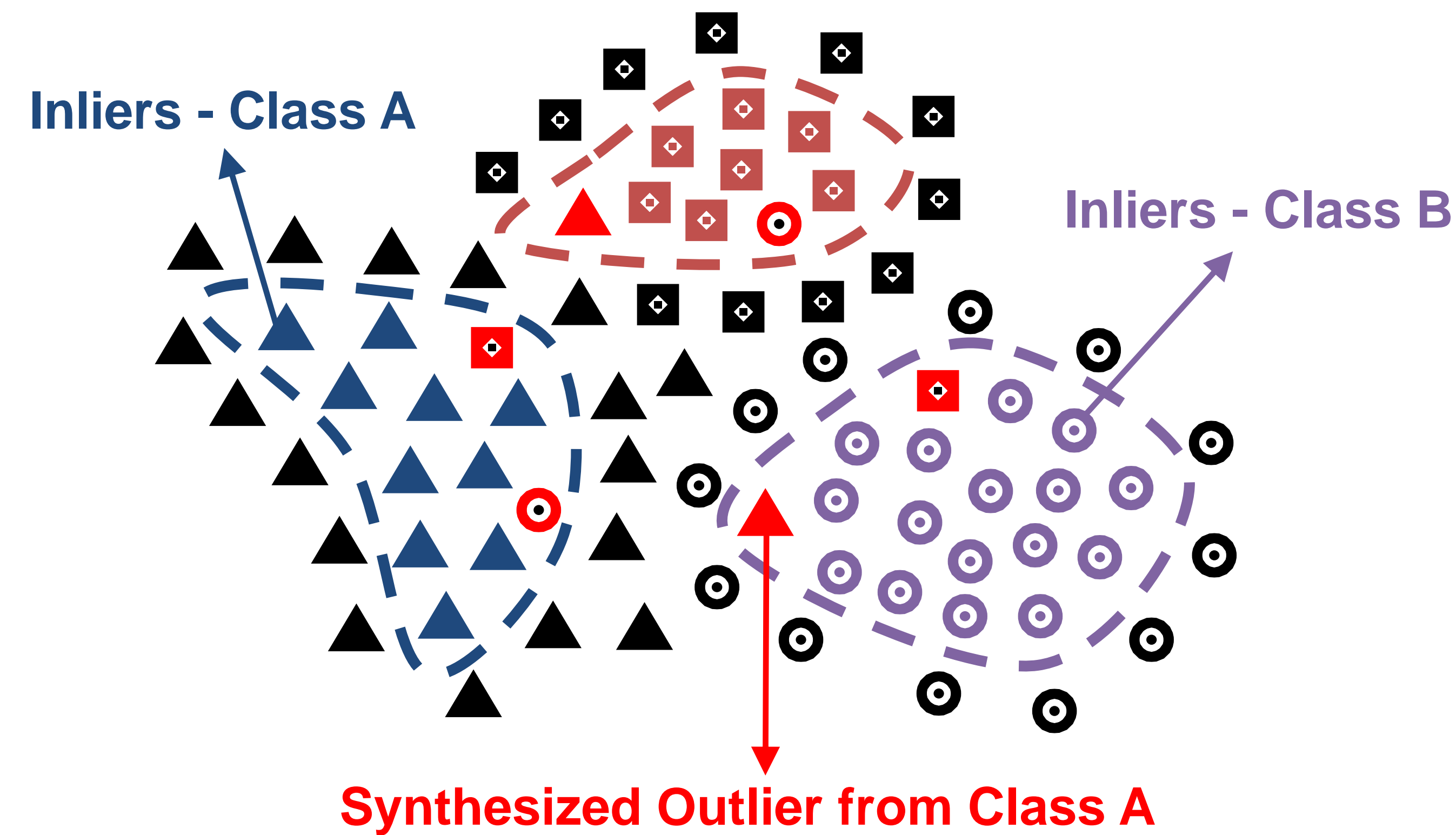
Flow Feature Synthesis

➤ Trains Normalizing Flow to map inlier object features to normal distribution latent space. Ensures a good estimation of the whole inlier distribution.


$\mathcal{L}_{det}$

$\mathcal{L}_{reg}$

Backbone → RPN → ROI Align → Box Head / Classifier → Energy

Feature Space $p(x)$ — Normalizing Flow Network $f_1\ f_2\ f_3 \dots f_{n-1}\ f_n$ $\mathcal{L}_{nll}$ — Latent Space $p(z) \sim \mathcal{N}(\mu, \Sigma)$

➤ Selects outlier features from the low-likelihood region of the synthetically generated features after random sampling from the latent space.

➤ Guarantees sampled outliers to be outlier with respect to all inlier classes.

VOS

FFS (ours)
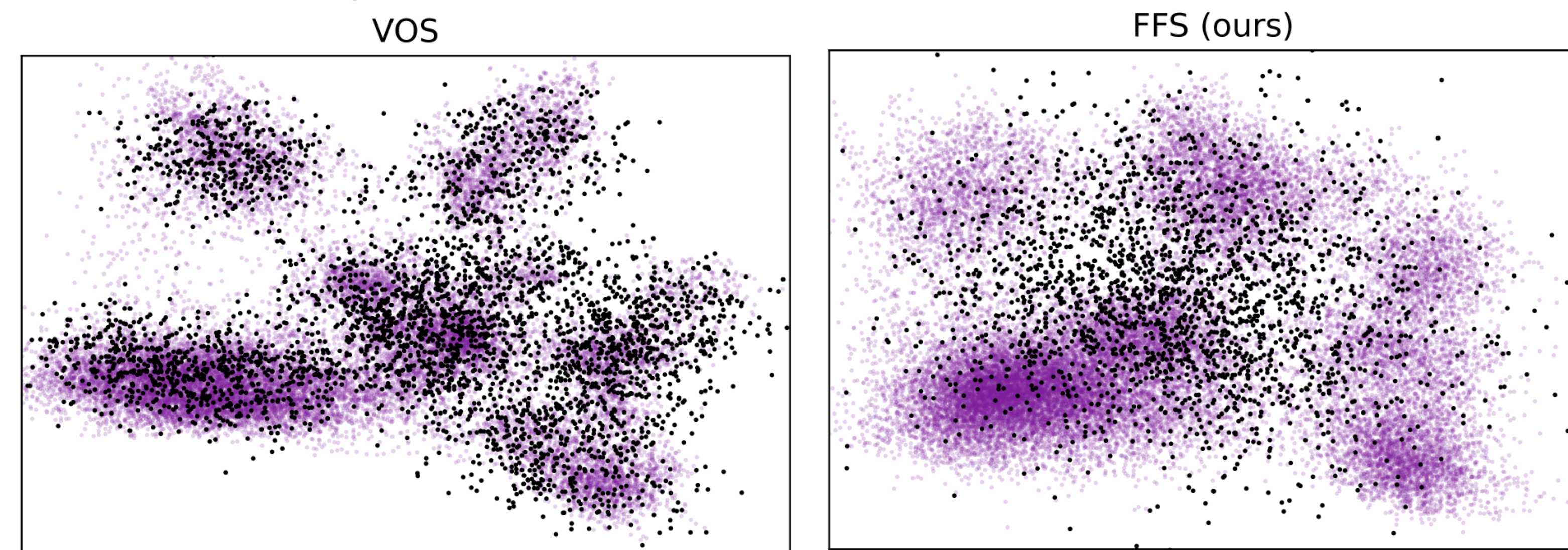
Results

➤ Our Normalizing Flow based approach better distinguishes the inlier, background and outlier distributions than VOS.

VOS

FFS (ours)
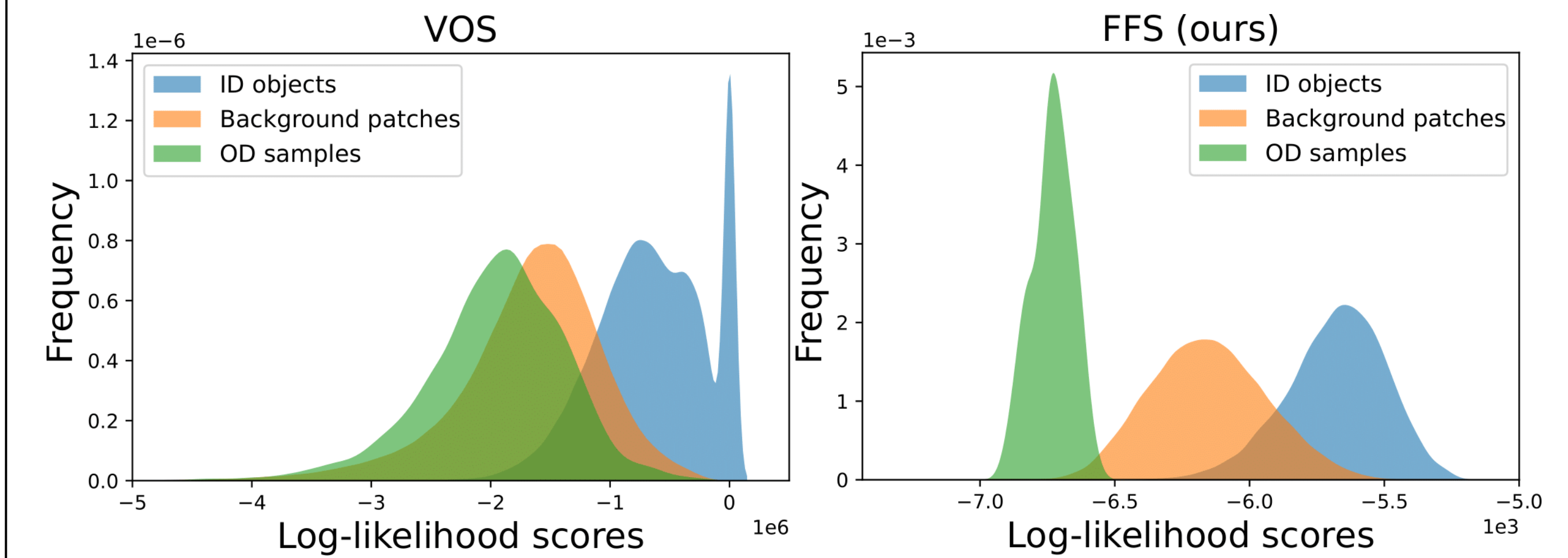
➤ Better in detecting outliers, reducing incorrect bounding boxes and decreasing the confidence on false inlier detection.

ID/OD	Method	FPR95	AUROC
VOC/COCO	VOS	47.77	89.00
	FFS	44.15	89.71
VOC/OpenImages	VOS	48.33	87.59
	FFS	45.08	88.29
BDD100K/nuImages	STUD	79.75	76.55
	FFS	76.68	77.53
Youtube-VIS/COCO	STUD	81.14	74.82
	FFS	83.06	76.37
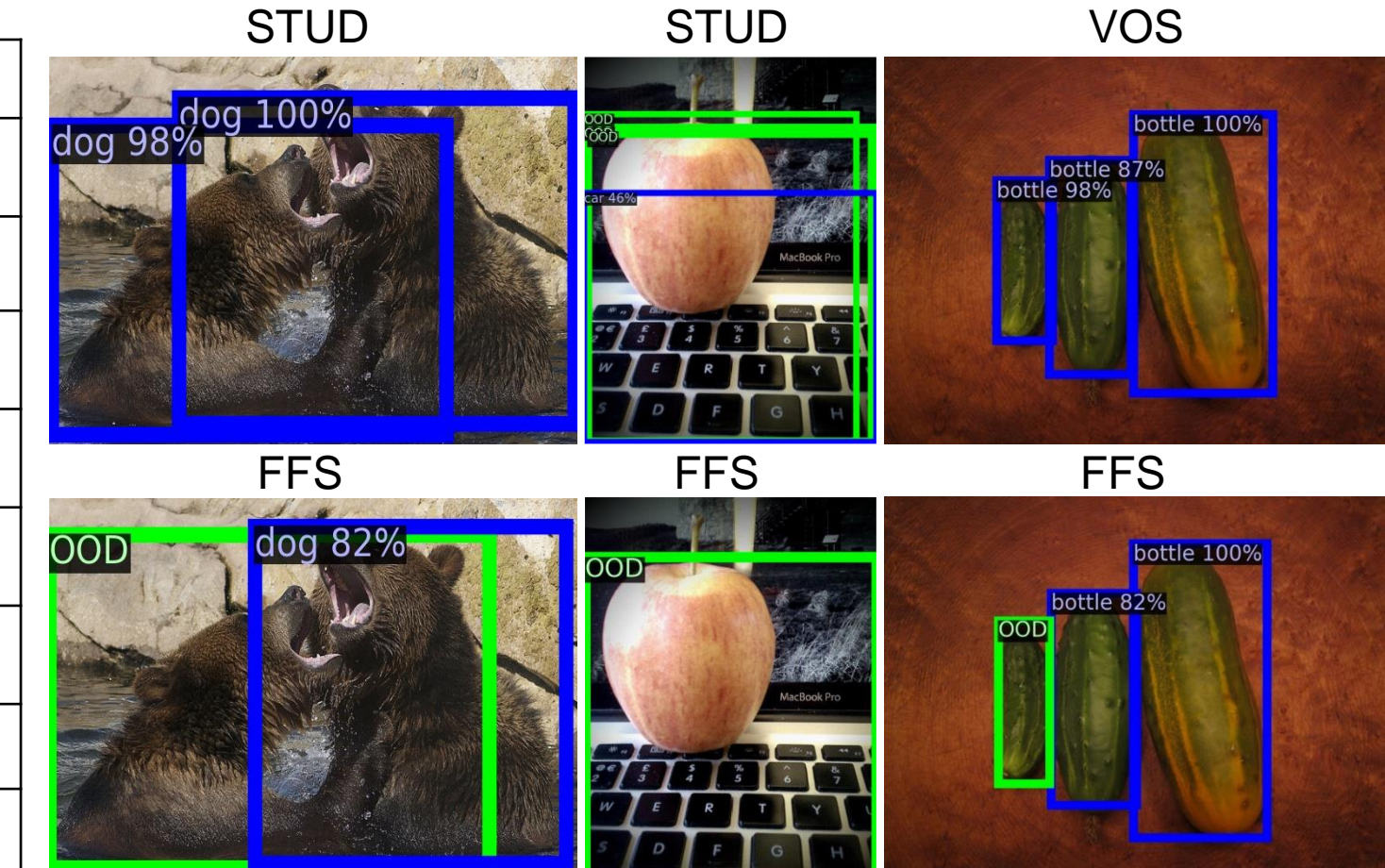
Conclusion

➤ Learns a combined data distribution from all inlier classes.

➤ Generates suitable outliers for training classification head.

➤ Achieves state-of-the-art for outlier-aware object detection.

➤ Scan QR Code for https://github.com/nish03/FFS.