Comprehensive Underwater Object Tracking Benchmark Dataset and Underwater Image Enhancement With GAN

Karen Panetta, *Fellow, IEEE*, Landry Kezebou, *Graduate Student Member, IEEE*, Victor Oludare, *Graduate Student Member, IEEE*, and Sos Agaian, *Fellow, IEEE*

***Abstract*—Current state-of-the-art object tracking methods have largely benefited from the public availability of numerous benchmark datasets. However, the focus has been on open-air imagery and much less on underwater visual data. Inherent underwater distortions, such as color loss, poor contrast, and underexposure, caused by attenuation of light, refraction, and scattering, greatly affect the visual quality of underwater data, and as such, existing open-air trackers perform less efficiently on such data. To help bridge this gap, this article proposes a first comprehensive underwater object tracking (UOT100) benchmark dataset to facilitate the development of tracking algorithms well-suited for underwater environments. The proposed dataset consists of 104 underwater video sequences and more than 74 000 annotated frames derived from both natural and artificial underwater videos, with great varieties of distortions. We benchmark the performance of 20 state-of-the-art object tracking algorithms and further introduce a cascaded residual network for underwater image enhancement model to improve tracking accuracy and success rate of trackers. Our experimental results demonstrate the shortcomings of existing tracking algorithms on underwater data and how our generative adversarial network (GAN)-based enhancement model can be used to improve tracking performance. We also evaluate the visual quality of our model's output against existing GAN-based methods using well-accepted quality metrics and demonstrate that our model yields better visual data.**

***Index Terms*—Underwater benchmark dataset, underwater generative adversarial network (GAN), underwater image enhancement (UIE), underwater object tracking (UOT).**

I. Introduction

UNDERWATER object tracking is pivotal in applications, such as underwater search and rescue operations, homeland and maritime security, deep ocean exploration, underwater robot navigation, and sea life monitoring [1]–[3]. These applications require efficient and accurate vision-based underwater sea analytics, including image enhancement, image quality assessment, and target tracking methods. On the other hand, high-noise and low-light situations pose enormous challenges for marine image/video analytics understanding. Further exacerbating these issues are the inherent underwater distortions including absorption and scattering of light causing low contrast, nonuniform illumination, diminished colors, and fuzz [4]. This makes computer vision tasks for detection, recognition, and tracking in underwater environments much more challenging than in open-air environments.

The recent success in object tracking has been facilitated by dedicated benchmarking datasets such as object tracking benchmark (OTB) [5], [6], visual object tracking (VOT) [7], and multiple object tracking (MOT) [8], [9]. Although several object tracking methods have been proposed over the years [10]–[16], the bias of the publicly available benchmark datasets, which mostly focus on open-air visual data, has greatly skewed the strength of these tracking algorithms to open-air environments. This is because the visual data these trackers are trained and tested on are not representative of underwater scenarios. As such, they each degrade in performance when tested on underwater scenarios as demonstrated in previous exploratory work [17]. This motivates the necessity to develop a comprehensive underwater database and benchmark to foster the development of tracking algorithms that will achieve comparatively high performance in both underwater and open-air environments.

Several methods have been proposed to overcome the inherent distortions in underwater data by enhancing image visual quality, including color restoration algorithms [18], contrast enhancement [19]–[21], quality metrics [21], [22], and deblurring [23]. Most recently, generative adversarial networks (GANs) have been used for automatic underwater image correction and enhancement [24], [25]. While other methods work well for correcting one type of distortion, GAN-based methods attempt to improve visual quality by translating visual data from the underwater domain to its equivalent "enhanced" domain. Existing GAN-based models have shown promising results on such translation tasks [24]–[28]. However, more work needs to be done to improve existing underwater GAN models, including addressing the weaknesses exhibited in either color, contrast or sharpness correction tasks. In this article, an efficient GAN model is developed to improve the performance of existing trackers on distorted underwater data by efficiently translating

Manuscript received January 30, 2020; revised November 12, 2020, April 22, 2021, and May 17, 2021; accepted May 27, 2021. Date of publication July 28, 2021; date of current version January 13, 2022. *(Landry Kezebou and Victor Oludare contributed equally to this work.) (Corresponding author: Landry Kezebou.)*

Associate Editor: J. Cobb.

Karen Panetta, Landry Kezebou, and Victor Oludare are with the Department of Electrical and Computer Engineering, Tufts University, Medford, MA 02155 USA (e-mail: karen@ece.tufts.edu; landry.kezebou@tufts.edu; victor.oludare@tufts.edu).

Sos Agaian is with the City University of New York, New York, NY 10017 USA (e-mail: sos.agaian@csi.cuny.edu).

Digital Object Identifier 10.1109/JOE.2021.3086907

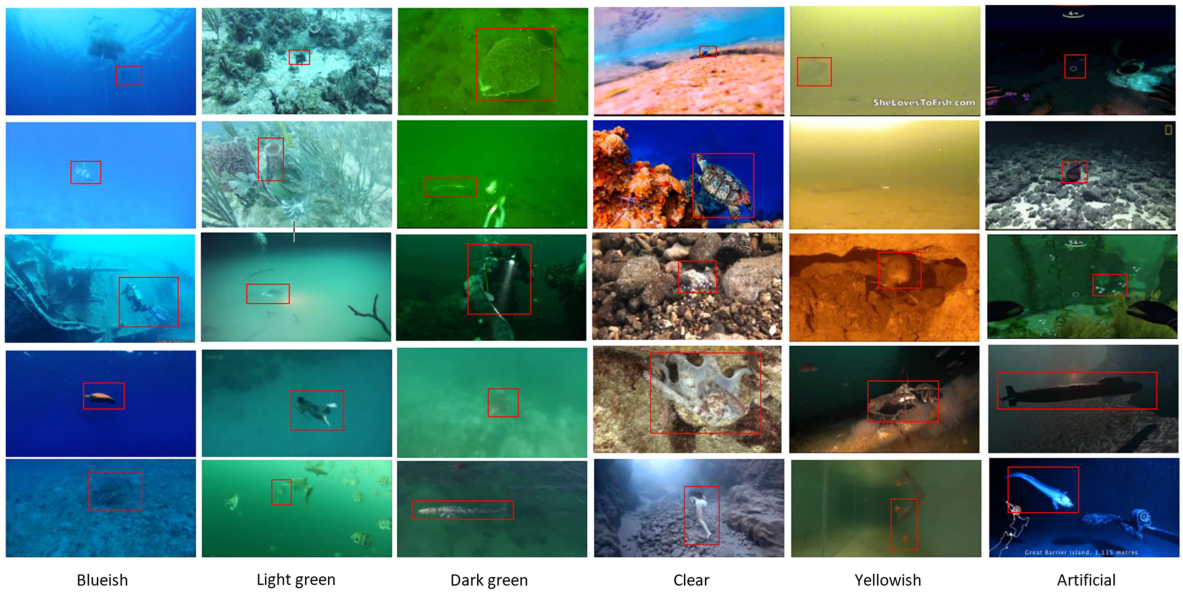


Fig. 1. Sample tracking data from our UOT100 dataset showing various types of distortions. The red bounding boxes denote the object of interest and the text below each column indicates the category of the visual data.

the distorted data to their non-distorted/enhanced or clear underwater versions. The proposed model also addresses the above-mentioned weaknesses in existing GAN models on underwater image enhancement tasks as demonstrated in Fig. 7.

Our main contributions in this article are as follows.

1) Creation of the first comprehensive and diversified underwater benchmarking dataset (UOT100), consisting of 104 video sequences and over 74 000 annotated frames. The distortions in the dataset are well distributed, including both artificial underwater and natural underwater visual data. The distortions are also summarized into visual categories that describe the type of water including blueish, greenish, and yellowish water visual data as shown in Fig. 1. *UOT100* is available for download on Kaggle at: https://www.kaggle.com/landrykezebou/uot100-underwater-object-tracking-dataset.
2) The performance of 20 object trackers is benchmarked on the entire dataset and analysis across distortion types are discussed.
3) Furthermore, a new cascaded residual network for underwater image enhancement (CRN-UIE), a GAN-based enhancement method for improving the trackers' performance on underwater data by translating visual data from the underwater domain to enhanced/clear underwater domain is presented.

The generator network in our architecture, as shown in Fig. 2, uses cascaded residual blocks and incorporates gradient profile (GP) loss for optimum enhancement. Dedicated underwater evaluation metrics, such as CCF [29] and UIQM [30], are used to compare performance of the CRN-UIE model with other state-of-the-art GAN models. Experimental results show that enhancing the visual data with CRN-UIE consistently yields much better tracking benchmark results as opposed to results on the raw data.

The remainder of this article is organized as follows: Section II gives a short background on object tracking algorithms, and reviews other OTB datasets. Next, existing underwater image enhancement methods are presented. Section III presents the detailed structure of the proposed dataset and evaluation metrics for benchmarking. Benchmark results of selected trackers on our dataset are presented in Section IV. In Section V, the CRN-UIE architecture and optimized loss functions are discussed, and improvement on the trackers' performance on the enhanced dataset are evaluated.

II. Background

A. OTB Datasets

Several object tracking datasets (OTBs) have been proposed in the past and are still being updated to include more challenging test data. The most popular benchmark datasets are competition datasets. OTB [5], [6], [31] is one of the most popular single object tracking challenge datasets. The earlier version of OTB, i.e., OTB50, consisted of 51 video sequences but was later updated to include up to 98 video sequences (OTB100) with more challenging tracking data. The sequences are classified into nine different attributes or categories, each representing a challenging aspect of visual tracking, including partially occluded targets, motion blur, deformation, fast motion, background clutter, out of view, in-plane rotation, scale variation, and illumination variation. A well-maintained online toolkit [31] is available for the user to run their trackers on the dataset and compare their performance to the most recent state-of-the-art tracking algorithms. OTB challenge uses success rate and precision plots in one pass evaluation (OPE) to evaluate the performance of trackers. This will be used as the evaluation metric for the presented benchmarking dataset.

Since 2013, the VOT challenge [7], [32] has published top-performing tracking algorithms on their competition dataset.

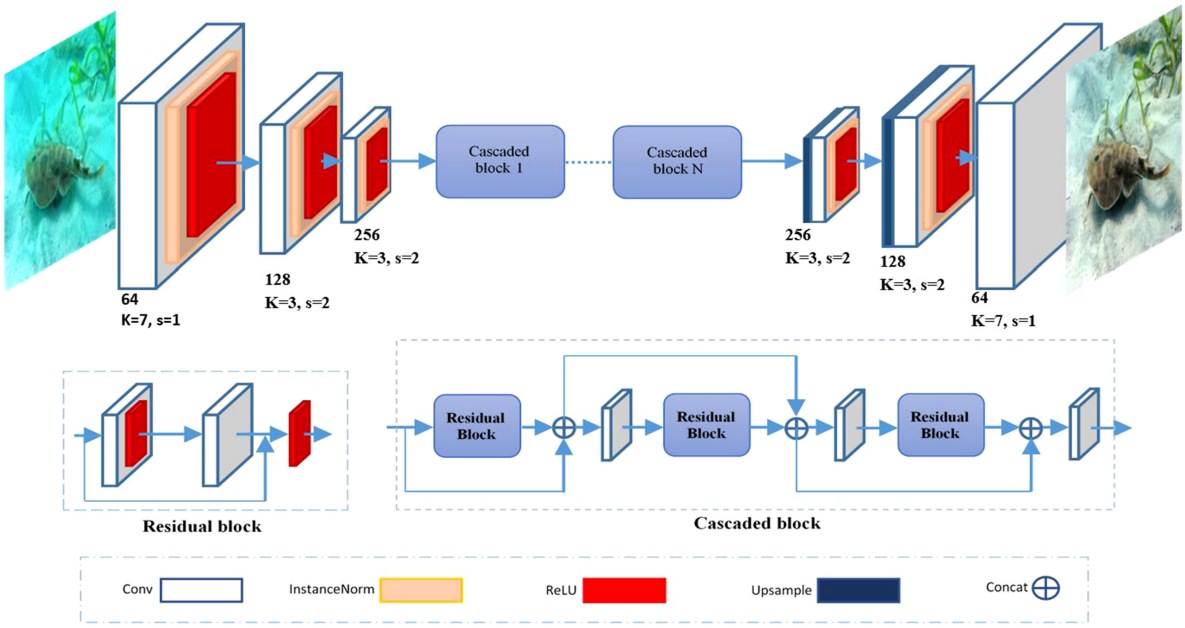


Fig. 2. Generator network architecture of the CRN-UIE. Conv denotes 2-D convolution layer and ReLU is the rectified linear unit activation function. The blocks are color coded and the numbers below each combo block represents the number of filters, kernel size, and stride size.

TABLE I
COMPARING EXISTING OBJECT TRACKING DATASETS TO UOT100

Benchmark Dataset	Open-air Visual data	Underwater Visual data	#Video sequences	#Annotated frames	Avg # frames / video
OTB50 [6], [31]	✓		51	29,491	578
OTB100 [5], [31]	✓		98	58,610	598
MOT16 [8]	✓		14	182,326	13,023
MOT17 [34]	✓		21	564,228	845
VOT (13, 14, 15, 16, 17, 18, 19) [7], [32]	✓		60	21,455	357
UAV123 [35]	✓		123	110,000	894
NUS-PRO [36]	✓		365	135,305	370
TLP (Long Term) [37], [38]	✓		50	676,000	13,520
Long-Term Tracking in the Wild [39], [40]	✓		337	1.55M	4,599
UOT32 (ours) [17]		✓	32	24,241	757
UOT100 (ours)		✓	**104**	**74,042**	**698**

The Thermal Infrared challenge of the VOT (VOT-TIR2015) encouraged the research community to develop state-of-art trackers with optimal performance on thermal images.

The MOT [8], [9] benchmark dataset is aimed at advancing research in the development of efficient MOT algorithms. Like VOT, MOT provides a yearly list of top-performing trackers on their 2-D and 3-D datasets. Evaluation is based on two metrics: accuracy and robustness. Accuracy measures the distance between tracker and ground truth bounding box centroids, whereas robustness measures the frequency of tracking failures [33].

The success of these competitions and other challenge benchmark datasets has led to numerous research publications in object tracking. Although existing benchmarks have strived to incorporate as many challenging aspects of visual tracking as possible, the inherent tracking challenges in the underwater visual domain is yet to be explored. Our goal is to make the dataset available and generate traction that will enable researchers to develop more sophisticated trackers for underwater environments. A brief comparison of UOT100 with some prominent challenge datasets is provided in Table I.

B. Object Tracking Algorithms

With the availability of large annotated datasets, numerous sophisticated tracking algorithms have been proposed [10]–[16], [41]–[51]. The methods used in most of these trackers can be

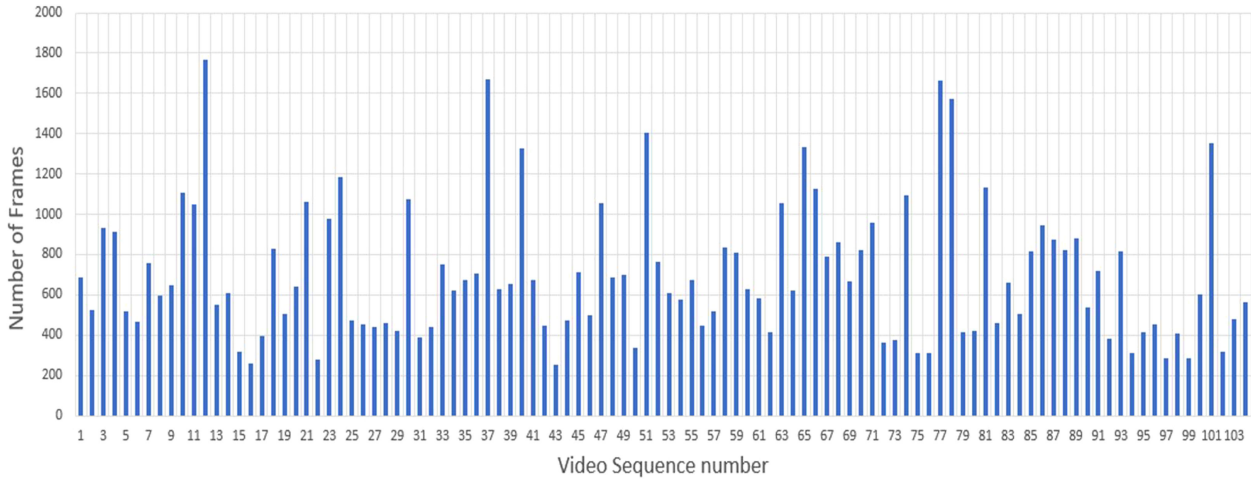


Fig. 3. Overview of number of annotated frames distribution across the UOT100 dataset.

classified into various categories depending on how features are extracted and used for tracking. Trackers such as KCF [10], HCF [47], DCF [44], [52], CFNet [51], STRCF [15], BACF [14], and fDSST[45] are considered kernelized correlation filters. The CNN feature-based approach has been predominantly used in the state-of-the-art trackers, such as MDNet [49], SiamFC [12], CCOT [48], and ECO [11]. Older trackers such as Struck [50] are based on local and global feature extraction.

C. Image Enhancement and Image Quality Measure

The need to enhance images in underwater environments is evident in various applications, such as underwater exploration, marine species tracking, and the use of unmanned underwater vehicles for wreckage inspection/recovery, where distortions affect the maneuverability of robots and tracking performance.

The breakthrough in Image-to-Image translation tasks using conditional GANs [53]–[56] has inspired the development of other dedicated GAN models for translating images from one domain to the other, including day to night, aerial photo to map, and edges to photo. GANs can be trained in a supervised manner on paired data or unsupervised manner if paired data is unavailable [57]. Li *et al.* [58] proposed WaterGAN, a model that generates realistic underwater images from open-air images and depth pairings in an unsupervised way for color correction of underwater images. Fabbri *et al.* [24] generated realistic pairings of training data by using CycleGAN [57], and then trained an underwater enhancement model based on the pix2pix architecture [55]. Guo *et al.* [28] proposed a multiscale dense GAN to boost the performance of the model by reusing previous features (residuals) and rendering more details leading to better enhanced images. Islam *et al.* [59] proposed a fast enhancement model based on the UNet architecture and an objective function that evaluates the perceptual quality of generated images based on its global content, color, and local style information.

Inspired by work in [24], [56], [59], and [28], we propose a network that generates high-quality, enhanced underwater outputs given distorted inputs. The proposed network replaces the residual blocks with cascaded residual blocks to use more rich features from previous layers. The loss function is also optimized to account for high-frequency information by adding the GP loss to the objective function.

III. UOT100 Dataset and Evaluation Metrics

A. UOT100 Dataset

Here, a diversified and comprehensive underwater object tracking (UOT100) benchmark dataset, consisting of 104 underwater video sequences sourced and segmented from different YouTube videos, is presented. These samples are diverse and vary by camera type, imaging conditions (viewing distance, viewing angle, background, and water quality), and target objects. The dataset also contains four videos of artificial underwater images that are generated from the *Subnautica*[1] game. This is to introduce more diversity and distortions in the dataset and test the performance of trackers in simulated underwater environments. The dataset contains a total of 74 042 annotated frames, with an average of 698 annotated frames and 26.2 s per video and captures a wide variety of underwater distortions. Samples of the underwater tracking data extracted from the UOT100 dataset are shown in Fig. 1, with each column showing variations of similar types of visual data. The UOT100 benchmark dataset is a substantially improved version over an earlier dataset, i.e., UOT32 [17]. UOT100 is a much larger dataset and includes many more underwater distortion categories. Here, a comprehensive analysis and a new enhancement algorithm that results in drastic performance increases of tracking algorithms are presented.

The dataset is organized in folders and subfolders. The root folder "*UOT100*" contains all the sequences in the dataset, and each sequence is stored in a separate folder. Each sequence folder contains an "*img*" folder, which includes all frames in the sequence; an mp4 video file; a ground truth text file that contains the ground truth annotations for each frame in the sequence; and a description file listing the distortion categories that the

[1][Online]. Available: https://unknownworlds.com/subnautica/

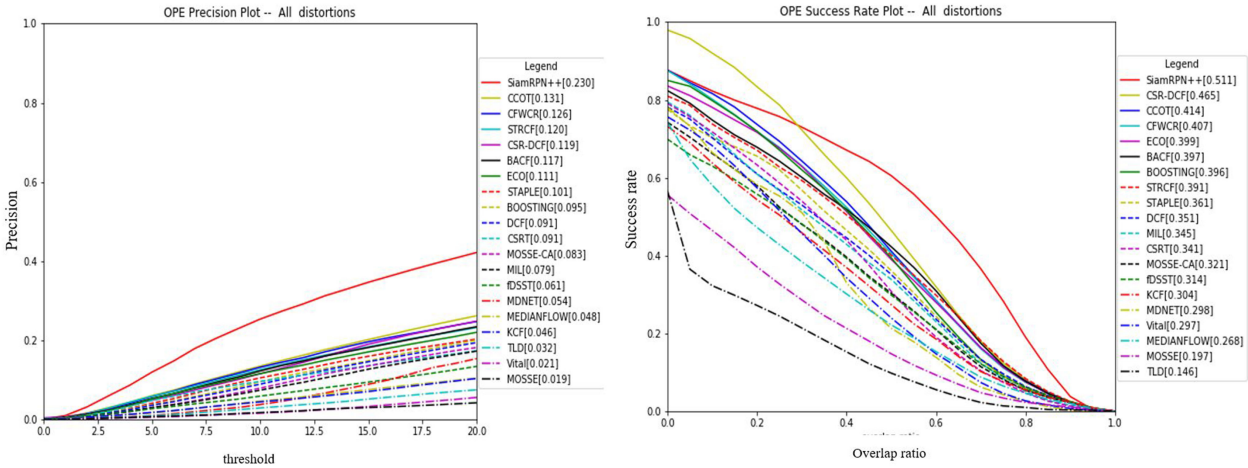


Fig. 4. Precision and success rate plots on the UOT100 dataset—entire dataset. In the legend, the values in brackets after each tracker represent the AUC of the corresponding plot. Trackers in the legend are ranked from top to least performing using their AUC values.

sequence is assigned to. Fig. 3 shows the distribution of the number of frames across the dataset.

B. Evaluation Metrics

The OPE protocol of the OTB benchmark proposed by Wu *et al.* [5] will be adopted in this work for evaluating trackers performance. The average precision and the success rate were computed to benchmark the performance of tracking algorithms on UOT100 dataset.

The precision is calculated as the distance error between the center pixel of the ground truth bounding box and that of the predicted bounding box. The prediction of the tracker for frame k is considered accurate if the pixel distance from predicted to the ground truth is less than a given pixel threshold. The precision plots shown in subsequent figures are computed by calculating the precision for each tracker at increasing threshold values between 0 and 20 pixels

$$P_k = \begin{cases} 1, \text{ if } \sqrt{\left(C^{gt} - C^{tk}\right)^2} \leq \text{Thres.} \\ 0, \text{ otherwise} \end{cases} \tag{1}$$

The average precision of a tracker over a sequence is calculated as the ratio of the total number of accurate predictions to the total number of frames in the sequence

$$P_{\text{seq}} = \frac{1}{N} \sum_{1}^{N} P_k. \tag{2}$$

Precision alone is not enough to compare the performance of multiple trackers on a given sequence or dataset. This is because precision does not consider how accurate the predicted bounding box is with respect to the ground truth bounding box in terms of size and how much area of the target object is covered in the predicted bounding box. The precision metric is therefore complemented by the success rate, which measures the Intersection over Union (IoU) representing the overlap ratio between the ground truth bounding boxes and the predicted bounding boxes, where a *0* ratio means no overlap (complete miss) and a *1* means 100% overlap (perfect match). The IoU is defined in the following equation and success rate plots are obtained by calculating IoU at various threshold values between *0* and *1*:

$$\text{S} = \frac{|\text{GT}_{\text{bbox}} \cap \text{TK}_{\text{bbox}}|}{|\text{GT}_{\text{bbox}} \cup \text{TK}_{\text{bbox}}|}$$

$$\begin{cases} GT_{\text{bbox}} : \text{ground truth bounding box} \\ TK_{\text{bbox}} : \text{tracker bounding box} \\ \cap \text{ and } \cup : \text{intersection and union, respectively.} \end{cases} \tag{3}$$

IV. Experiment and Benchmarking the UOT100 Dataset

The performance of 20 state-of-the-art tracking algorithms using the UOT100 dataset was obtained. Some of the trackers are from the OTB benchmark, and others are more recent deep learning-based tracking algorithms including BACF [14], BOOSTING [60], CCOT [48], CFWCR [61], CSR-DCF [52], DCF [44], ECO [11], fDSST [45], KCF [10], MDNet [49], MEDIANFLOW [62], MIL [42], MOSSE [63], SiamMask [64], SiamRPN++ [65], STAPLE [16], STRCF [15], TLD [66], and VITAL [67].

Next, the performance of the selected trackers across the entire UOT100 dataset was investigated and reported using the OPE precision and success rates. The area under curve (AUC) is used to further rank the performance of the trackers listed. Fig. 4 shows the OPE precision and success rate benchmark plots across the entire dataset. The deep learning method, i.e., SiamRPN++, achieves the highest performance on the UOT100 dataset, in both precision and success rate, as shown by the red curve. However, while SiamRPN++ did well on natural open-air datasets like OTB, it exhibited significant performance degradation on underwater data, achieving only a precision AUC of *0.230* and a success rate AUC of *0.511* on the UOT100, as opposed to a precision and success rate AUC of *0.914* and *0.696*, respectively, on the OTB.

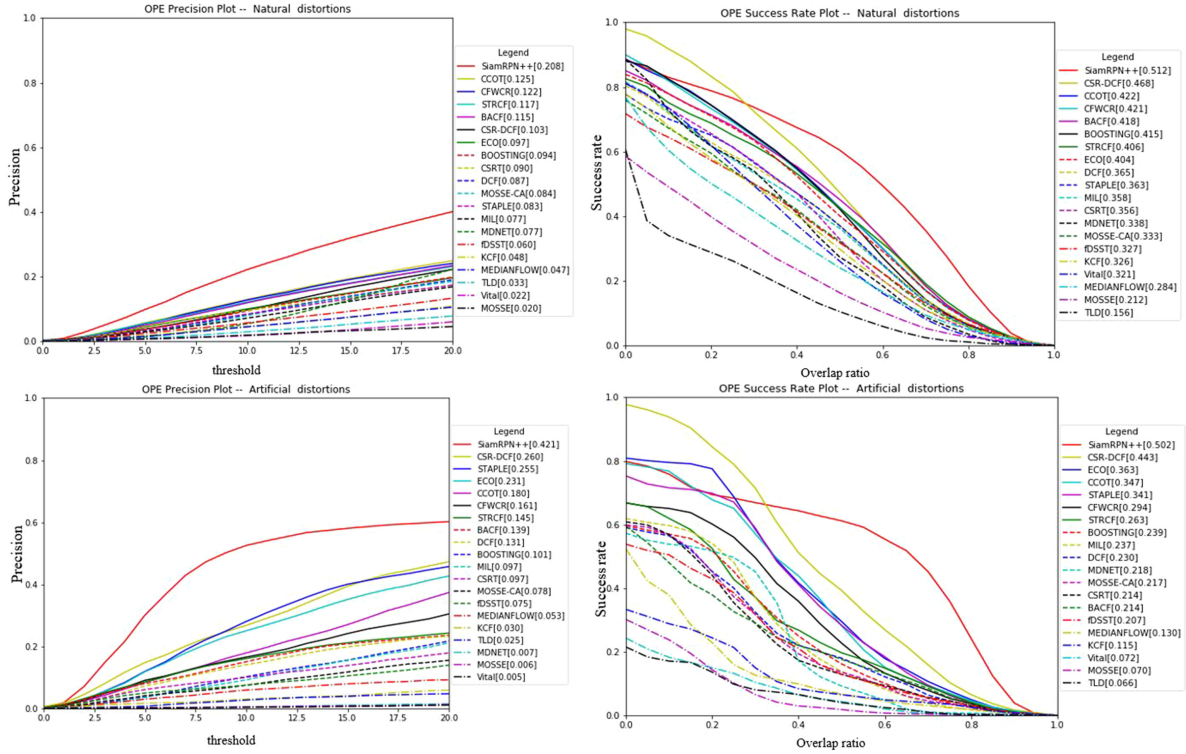


Fig. 5. OPE precision and success rate benchmark plots on natural (first row) versus artificial underwater (second row) visual data. Trackers in the legend are ranked from top to least performing using their AUC values.

CCOT, CFWCR, STRCF, CSR-DCF, ECO, BACF, and STAPLE all exhibited low performance on the UOT100 with precision of *0.131, 0.126, 0.120, 0.119, 0.111, 0.117,* and *0.101,* and success rate AUC of *0.511, 0.407, 0.391, 0.465, 0.399, 0.397,* and *0.361,* respectively. Other trackers never reached a precision AUC above *0.1*, demonstrating the difficulty of tracking in an underwater environment.

Analyzing further, the trackers' performance across the different types of underwater visual qualities including natural versus artificial, and clear versus distorted can be seen in Figs. 5 and 6.

Fig. 5 compares the benchmark performance of the trackers on natural versus artificial underwater visual data, while the comparison of trackers performance on clear underwater versus distorted underwater visual data is shown in Fig. 6. The trackers tend to exhibit slightly better precision AUC performance on artificial underwater visual data than natural. However, the corresponding success rate plots for artificial data are much less stable. This is because artificially generated underwater data either undersimulate or oversimulate distortions and often do not capture them all. Further analysis based on different visual distortion categories are included in Appendix A. The trackers also tend to perform better on clear underwater than distorted underwater visual data, especially looking at the AUC results on the OPE success rate plots.

SiamRPN++, CFWCR, CCOT, ECO, and STRCF trackers consistently ranked as the best trackers on the overall dataset and across simulations on various types of underwater visual data and distortions.

Table II reports the AUC performance of the trackers across the UOT100 dataset and under various types of underwater distortions and visual quality. This represents the data used in generating the benchmark plots. The highlighted blue values indicate the best performance among all benchmarked trackers for each evaluation set.

V. Proposed CRN-UIE Model

The foregoing section demonstrated the shortcomings of object tracking algorithms on underwater visual data. In this section, a new GAN-based method called CRN-UIE aimed at improving the performance of trackers on underwater data is shown. We discuss the details of the proposed model and the optimized loss function. The merits of CRN-UIE are demonstrated by comparing its enhanced outputs to the outputs of other dedicated underwater GANs. Finally, CRN-UIE is used to enhance the visual quality of the UOT100 dataset, and subsequently, the performance of the selected trackers is again benchmarked on the enhanced dataset to help visualize improvement in tracking performance in underwater environments.

A. Generator Loss Function Optimization

Given a distorted underwater image X, the goal is to learn functions that maps X to a target non-distorted/enhanced domain Y.

Inspired by the pix2pixHD network architecture [56], a modified generator network is proposed as illustrated in Fig. 2. The architecture uses three convolution layers for the encoding

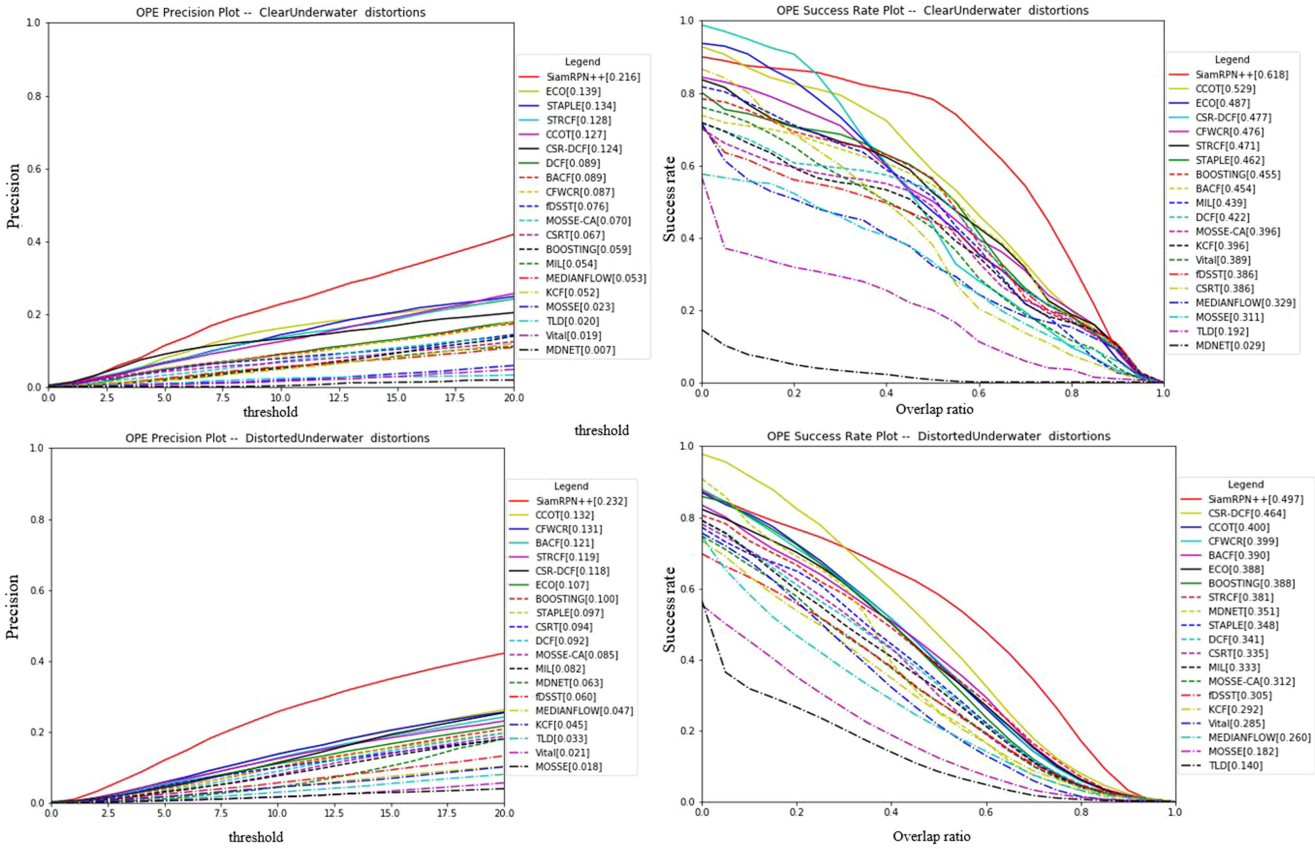


Fig. 6. OPE precision and success rate benchmark plots on clear (first row) versus distorted (second row) underwater visual data. Trackers in the legend are ranked from top to least performing using their AUC values.

TABLE II
AUC RESULTS FOR ALL TRACKERS ACROSS VARIATIONS OF UNDERWATER VISUAL QUALITY

Precision AUC (Area Under Curve)																				
	BACF	BOOSTING	CCOT	CFWCR	CSR-DCF	CSRT	DCF	ECO	fDSST	KCF	MDNET	MEDIANFLOW	MIL	MOSSE	MOSSE-CA	SiamRPN++	STAPLE	STRCF	TLD	Vital
All	0.117	0.095	0.131	0.126	0.119	0.091	0.091	0.111	0.061	0.046	0.054	0.048	0.079	0.019	0.083	0.23	0.101	0.12	0.032	0.021
Natural	0.115	0.094	0.125	0.122	0.103	0.09	0.087	0.097	0.06	0.048	0.077	0.047	0.077	0.02	0.084	0.208	0.083	0.117	0.033	0.022
Artificial	0.139	0.101	0.18	0.161	0.26	0.097	0.131	0.231	0.075	0.03	0.007	0.053	0.097	0.006	0.078	0.421	0.255	0.145	0.025	0.005
ClearUnderwater	0.089	0.059	0.127	0.087	0.124	0.067	0.089	0.139	0.076	0.052	0.007	0.053	0.054	0.023	0.07	0.216	0.134	0.128	0.02	0.019
DistortedUnderwa	0.121	0.1	0.132	0.131	0.118	0.094	0.092	0.107	0.06	0.045	0.063	0.047	0.082	0.018	0.085	0.232	0.097	0.119	0.033	0.021
BlueLike	0.156	0.12	0.174	0.178	0.146	0.137	0.112	0.154	0.082	0.046	0.074	0.06	0.117	0.026	0.099	0.279	0.123	0.167	0.048	0.025
DarkBlue	0.129	0.083	0.131	0.14	0.115	0.09	0.08	0.083	0.076	0.034	0.008	0.051	0.082	0.011	0.08	0.245	0.072	0.126	0.042	0.03
LightBlue	0.192	0.17	0.233	0.231	0.192	0.204	0.155	0.253	0.089	0.062	0.088	0.072	0.165	0.048	0.124	0.327	0.193	0.226	0.057	0.017
GreenLike	0.13	0.111	0.145	0.146	0.127	0.104	0.104	0.131	0.058	0.052	0.063	0.045	0.086	0.023	0.092	0.253	0.121	0.13	0.032	0.021
DarkGreen	0.152	0.089	0.152	0.138	0.138	0.069	0.113	0.108	0.07	0.045	0.061	0.058	0.059	0.011	0.083	0.26	0.109	0.13	0.033	0.023
LightGreen	0.116	0.123	0.141	0.151	0.119	0.126	0.099	0.146	0.05	0.056	0.063	0.037	0.102	0.03	0.097	0.248	0.129	0.131	0.031	0.019
YellowLike	0.067	0.079	0.102	0.146	0.19	0.02	0.166	0.114	0.016	0.011		0.027	0.035	0.047	0.138	0.248	0.013	0.024	0.008	0.009
Success rate AUC (Area Under Curve)																				
All	0.397	0.396	0.414	0.407	0.465	0.341	0.351	0.399	0.314	0.304	0.298	0.268	0.345	0.197	0.321	0.511	0.361	0.391	0.146	0.297
Natural	0.418	0.415	0.422	0.421	0.468	0.356	0.365	0.404	0.327	0.326	0.338	0.284	0.358	0.212	0.333	0.512	0.363	0.406	0.156	0.321
Artificial	0.214	0.239	0.347	0.294	0.443	0.214	0.23	0.363	0.207	0.115	0.218	0.13	0.237	0.07	0.217	0.502	0.341	0.263	0.066	0.072
ClearUnderwater	0.454	0.455	0.529	0.476	0.477	0.386	0.422	0.487	0.386	0.396	0.029	0.329	0.439	0.311	0.396	0.618	0.462	0.471	0.192	0.389
DistortedUnderwa	0.39	0.388	0.4	0.399	0.464	0.335	0.341	0.388	0.305	0.292	0.351	0.26	0.333	0.182	0.312	0.497	0.348	0.381	0.14	0.285
BlueLike	0.391	0.37	0.397	0.399	0.453	0.326	0.337	0.403	0.309	0.267	0.194	0.257	0.332	0.169	0.283	0.504	0.341	0.389	0.143	0.268
DarkBlue	0.394	0.352	0.377	0.383	0.469	0.302	0.311	0.348	0.3	0.261	0.181	0.244	0.311	0.144	0.25	0.48	0.289	0.372	0.108	0.294
LightBlue	0.387	0.396	0.425	0.422	0.43	0.36	0.373	0.481	0.321	0.276	0.196	0.273	0.361	0.207	0.327	0.537	0.412	0.414	0.19	0.232
GreenLike	0.382	0.396	0.404	0.404	0.457	0.342	0.351	0.405	0.295	0.293	0.335	0.242	0.338	0.193	0.331	0.515	0.362	0.386	0.142	0.282
DarkGreen	0.407	0.358	0.395	0.375	0.496	0.342	0.337	0.391	0.295	0.264	0.527	0.236	0.284	0.134	0.292	0.502	0.354	0.387	0.14	0.245
LightGreen	0.367	0.418	0.41	0.421	0.433	0.341	0.36	0.413	0.295	0.311	0.252	0.246	0.369	0.228	0.354	0.523	0.367	0.386	0.143	0.304
YellowLike	0.385	0.47	0.472	0.509	0.54	0.296	0.466	0.389	0.249	0.3		0.31	0.406	0.332	0.585	0.55	0.305	0.28	0.147	0.259

and decoding networks. The cascaded blocks ensure that features from previous layers are transferred to the subsequent layers, hence, high-frequency information from the encoded maps are preserved and properly decoded, generating a sharp enhanced image.

CRN-UIE also uses a multiscale discriminator similar to that of pix2pixHD. This consists of two identical discriminators D_1 and D_2. The output from the generator is fed to D_1, downsampled by a factor of 2, then fed to D_2. This process guides the generator to generate images that are both globally consistent and produce fine details. To optimize the model, we derive our generator loss function as follows.

1) *Adversarial loss* [68]: The generator tries to learn a mapping function $G : X \rightarrow Y$ and fool discriminators D_1 and

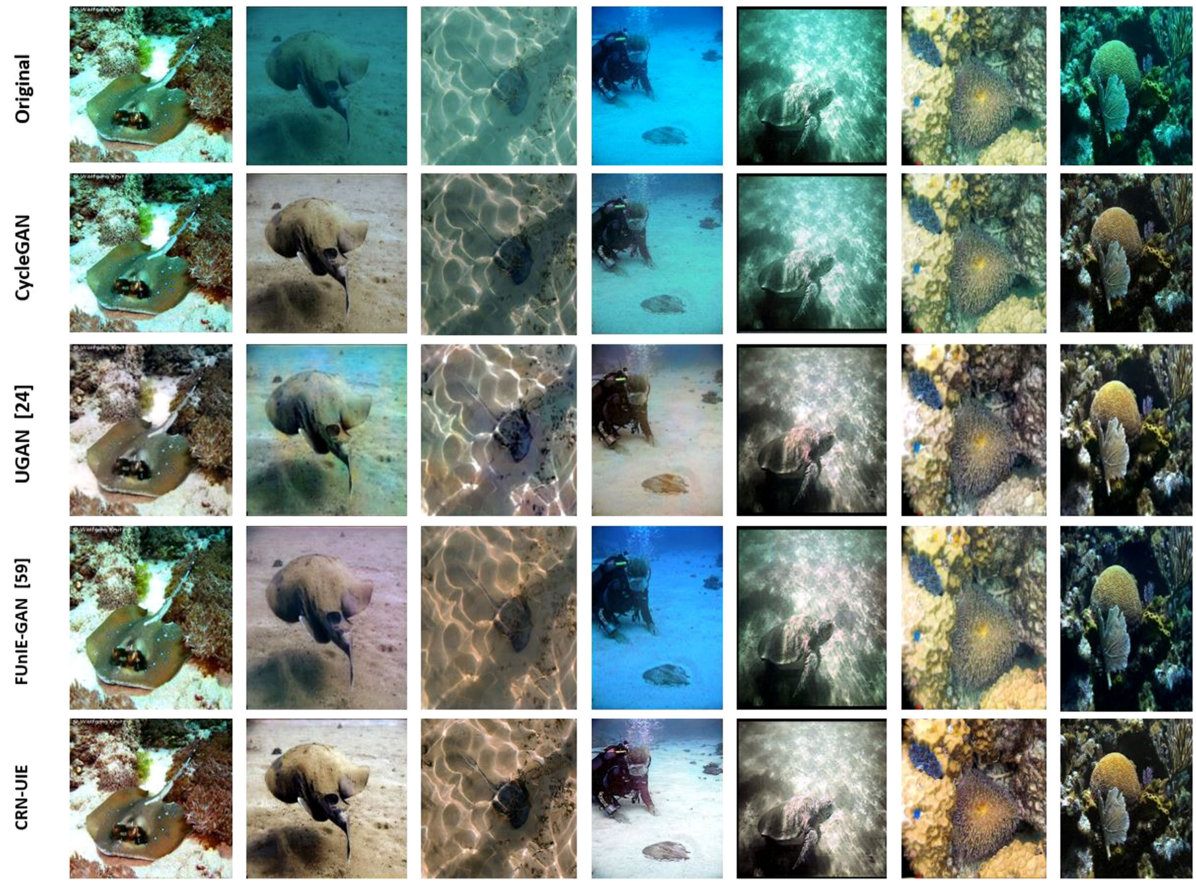


Fig. 7. Comparing the proposed CRN-UIE (ours) output with other state-of-the-art GAN models. CRN-UIE produces more realistic enhanced images from original distorted underwater input images. CRN-UIE yields better visual quality and better color restoration than other models, as substantiated quantitatively with data in Table III.

D_2. The adversarial loss is expressed as follows:

$$L_{\mathrm{ADV}} = \min_{G}\left[\max_{D_1,D_2}\sum_{k=12}\left[E_{(X)}\left[\log D_k\left(X\right)\right] + E_{(X)}\left[\log\left(1 - D_k\left(G\left(\hat{X}\right)\right)\right)\right]\right]\right] \quad (4)$$

where X and $\hat{X}$ denote distorted and undistorted underwater images, respectively, $G(\hat{X})$ is the output of the generator network, and k index differentiates between discriminators.

2) *Feature matching loss* [56]: This improves the adversarial loss by stabilizing training and ensuring that the generator produces reasonable statistical information at multiple scales. To do this, we learn to match intermediate feature maps between the real and generated image

$$L_{FM} = \min_{G}\left[\sum_{k\,=\,12} E_{(X)}\sum_{1=1}^{T}\frac{1}{N_i}\left[\left\|D_k^{(i)}\left(X\right) - D_k^{(i)}\left(G\left(\hat{X}\right)\right)\right\|_1\right]\right] \quad (5)$$

where T is the total number of layers, N_i is the number of elements in each layer, and $D_k^{(i)}$ is the ith-layer feature extractor of discriminator $\mathrm{D_k}$.

3) *Perceptual loss* [69]: This is used to measure the high-level perceptual and semantic differences between images. The activations of the jth layers of a pretrained VGG network for image classification denoted by $\phi_j()$ is extracted. Pixelwise distance is used to measure the difference between the perceptual features of the distorted and enhanced underwater image

$$L_{\mathrm{VGG}}^{\phi,j} = \frac{1}{C_j H_j W_j}\left\|\phi_j\left(\hat{Y}\right) - \phi_j\left(X\right)\right\|_1 \quad (6)$$

where $\hat{Y}$ is the generated image, H_j and W_j are the height and width of the jth feature map, and C_j indicates the channel.

4) *GP loss* [70]: Measures the difference between the edge information of the generated and target images

$$L_{\mathrm{GPL}}\left(X,\,\hat{Y}\right) = \sum_{c}\left(\frac{1}{H}\mathrm{trace}\left(\nabla G\left(\hat{Y}\right)_c\cdot\nabla X_c^{\tau}\right)\right.$$

TABLE III
QUANTITATIVE COMPARISON OF CRN-UIE MODEL VERSUS OTHER STATE-OF-THE-ART GAN MODELS USING WELL ACCEPTED NO-REFERENCE IMAGE QUALITY METRICS

Method	UICM	UISM	UIConM	UIQM	CCF
Original	-59.6120	5.6595	0.7320	2.6074	28.6571
CycleGAN [57]	-25.5348	6.4704	0.8401	4.1942	22.9768
FUnIE-GAN [59]	-14.9349	7.2558	0.8623	4.8044	26.0130
UGAN [24]	-3.1183	7.1064	0.8497	5.0485	27.6772
CRN-UIE(Ours)	**2.5228**	**6.5798**	**0.7842**	**4.8177**	**35.5643**

Implementations for CCF and UIQM evaluation metrics are available on the respective GitHub repo:CCF: https://github.com/zhenglab/CCF; UIQM: https://github.com/paulpanwang/hikvision

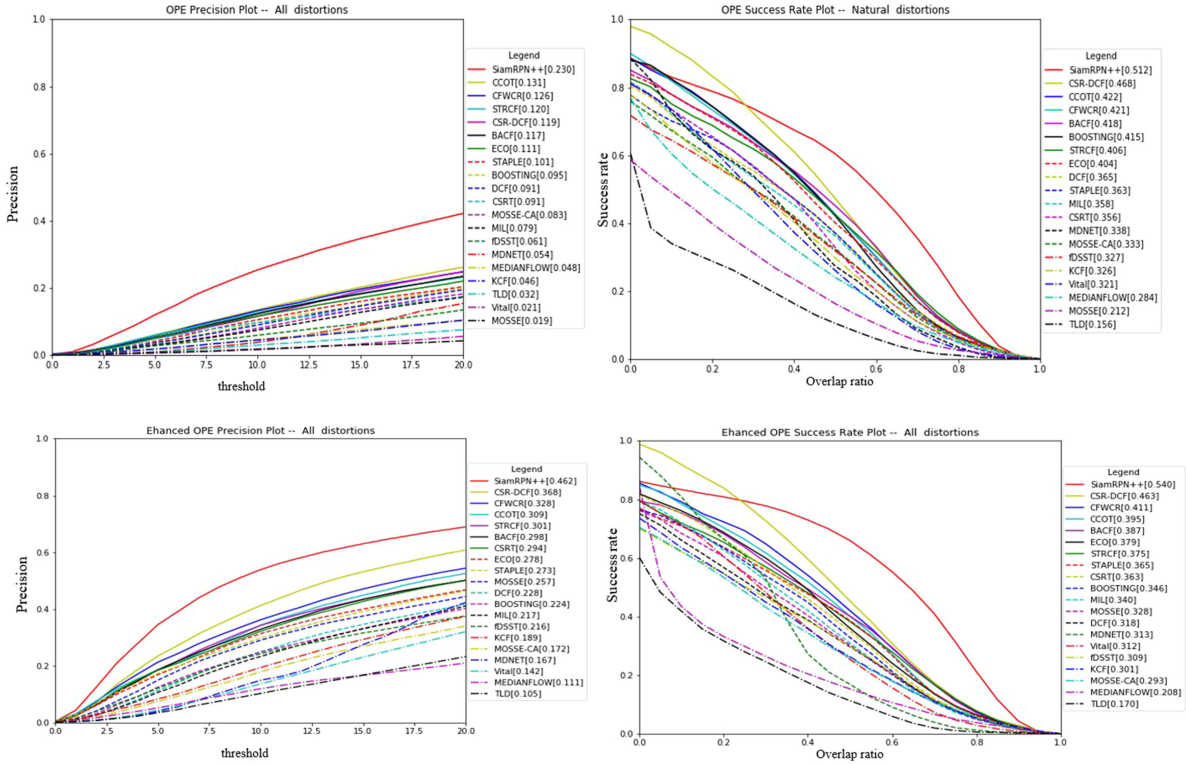


Fig. 8. Comparative benchmark results on original (first row) versus enhanced dataset (second row). Trackers in legend are ranked from top to least performing using their AUC values.

$$+\frac{1}{W}\text{trace}\left(\nabla G\left(\hat{Y}\right)_c^{\tau}\cdot\nabla X_c\right)\Bigg) \tag{7}$$

where $(\cdot)^{\tau}$ represents transpose, H and W are the height and width of the image, X is the target image, and $\hat{Y}$ is the generated image.

The total loss is therefore the sum of the adversarial loss L_{ADV}, feature matching loss L_{FM}, perceptual loss L_{VGG}, and GP loss L_{GPL}.

Enhancement results on the entire 1381 test images will be made available on our GitHub repo and Kaggle

$$L_{\text{CRN-UIE}} = L_{\text{ADV}} + \lambda_1 L_{\text{FM}} + \lambda_2 L_{\text{VGG}} + \lambda_3 L_{\text{GPL}} \tag{8}$$

where λ_1, λ_2, and λ_3 are constants. We set $N = 2$ where N is the number of cascaded blocks used. An Adam optimizer with an initial learning rate of 2e-4, beta1 = 0.5, and beta2 = 0.999 is used to optimize the loss function. The values were determined empirically. An intuitive and empirical hyperparameter search through extensive computer simulations indicated that $\lambda_1 = 10$, $\lambda_2 = 1$, and $\lambda_3 = 1$ represented the best parameter combination for the proposed model yielding the desired outputs. A *step decay* annealing strategy is used with a decay constant of 100 to decrease learning rate after 100 epochs.

Fig. 7 shows the comparative visual outputs of the CRN-UIE model compared with several other state-of-the-art models. Subjectively, CRN-UIE produces a better visual output on distorted to enhanced underwater image translation task. These results are further quantitatively substantiated in Table III. All four models being compared in Fig. 7, including the proposed CRN-UIE, were trained on the enhancing underwater visual perception

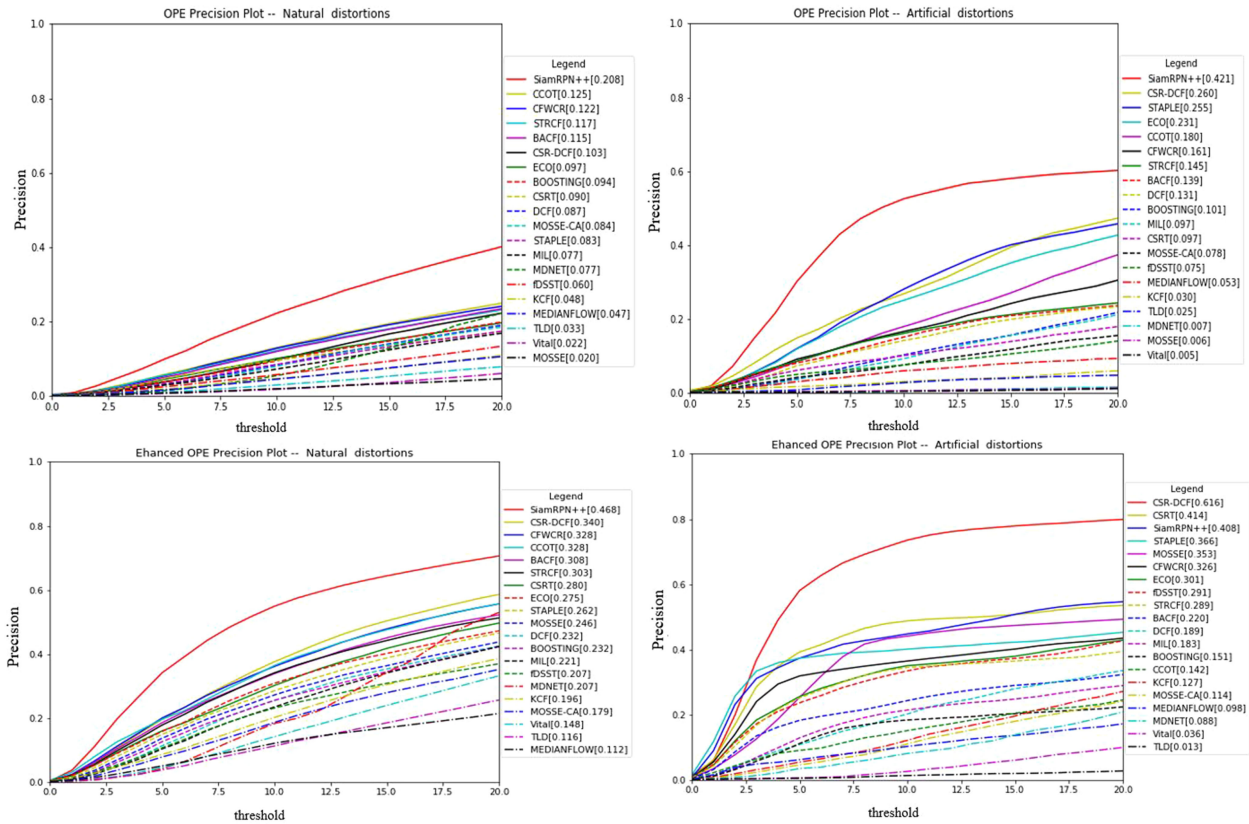


Fig. 9. Comparing the precision benchmark plots on original (first row) versus enhanced (second row) data (natural versus artificial underwater data). Trackers in legend are ranked from top to least performing using their AUC values.

TABLE IV
COMPARATIVE TRACKING BENCHMARK RESULTS ON DISTORTED AND ENHANCED UNDERWATER DATA

		Precision AUC (Area Under Curve)																			
		BACF	BOOSTING	CCOT	CFWCR	CSR-DCF	CSRT	DCF	ECO	fDSST	KCF	MDNET	MEDIANFLOW	MIL	MOSSE	MOSSE-CA	SiamRPN++	STAPLE	STRCF	TLD	Vital
Non-enhanced	All	0.117	0.095	0.131	0.126	0.119	0.091	0.091	0.111	0.061	0.046	0.054	0.048	0.079	0.019	0.083	0.23	0.101	0.12	0.032	0.021
	Natural	0.115	0.094	0.125	0.122	0.103	0.09	0.087	0.097	0.06	0.048	0.077	0.047	0.077	0.02	0.084	0.208	0.083	0.117	0.033	0.022
	Artificial	0.139	0.101	0.18	0.161	0.26	0.097	0.131	0.231	0.075	0.03	0.007	0.053	0.097	0.006	0.078	0.421	0.255	0.145	0.025	0.005
After GAN enhancement	All	**0.298**	**0.224**	**0.309**	**0.328**	**0.368**	**0.294**	**0.228**	**0.278**	**0.216**	**0.189**	**0.167**	**0.111**	**0.217**	**0.257**	**0.172**	**0.462**	**0.273**	**0.301**	**0.105**	**0.142**
	Natural	**0.308**	**0.232**	**0.328**	**0.328**	**0.34**	**0.28**	**0.232**	**0.275**	**0.207**	**0.196**	**0.207**	**0.112**	**0.221**	**0.246**	**0.179**	**0.468**	**0.262**	**0.303**	**0.116**	**0.148**
	Artificial	**0.22**	**0.151**	**0.142**	**0.326**	**0.616**	**0.414**	**0.189**	**0.301**	**0.291**	**0.127**	**0.088**	**0.098**	**0.183**	**0.353**	**0.114**	**0.408**	**0.366**	**0.289**	**0.013**	**0.036**

		Success rate AUC (Area Under Curve)																			
Non-enhanced	All	**0.397**	**0.396**	**0.414**	0.407	**0.465**	0.341	**0.351**	**0.399**	**0.314**	**0.304**	0.298	**0.268**	0.345	0.197	**0.321**	0.511	0.361	**0.391**	0.146	0.297
	Natural	**0.418**	**0.415**	**0.422**	0.421	0.468	0.356	**0.365**	**0.404**	**0.327**	0.326	0.338	**0.284**	0.358	0.212	**0.333**	0.512	0.363	0.406	0.156	0.321
	Artificial	**0.214**	**0.239**	**0.347**	0.294	**0.443**	0.214	**0.23**	**0.363**	**0.207**	**0.115**	0.218	0.13	**0.237**	0.07	**0.217**	**0.502**	**0.341**	**0.263**	**0.066**	0.072
After GAN enhancement	All	0.387	0.346	0.395	**0.411**	0.463	**0.363**	0.318	0.379	0.309	0.301	**0.313**	0.208	**0.346**	**0.328**	0.293	**0.54**	**0.365**	0.375	**0.17**	**0.312**
	Natural	0.412	0.371	0.417	**0.43**	**0.471**	**0.373**	0.34	0.396	0.322	**0.326**	**0.267**	0.219	**0.363**	**0.341**	0.319	**0.565**	**0.376**	**0.41**	**0.188**	**0.322**
	Artificial	0.176	0.13	0.194	0.25	0.39	**0.279**	0.13	0.227	0.191	0.103	**0.404**	**0.116**	0.138	**0.221**	0.074	0.326	0.262	0.164	0.017	**0.158**

(EUVP)[2] dataset prepared by the Interactive Robotics and Vision Lab from the University of Minnesota, Twin Cities, Minneapolis, MN, USA. The dataset was introduced in the fast underwater image enhancement (FUnIE) GAN in [59] and contains paired and unpaired image samples of clear and distorted underwater visual quality. FUnIE-GAN used the procedure in [24] to generate paired dataset. Here, for CRN-UIE, we use paired data from the EUVP dataset for training. We refer any reader to [24] for more detailed intuitions about using CycleGAN to generate paired data when obtaining natural pairs is unattainable. A quantitative evaluation of the performance of these models using no-reference measures including underwater image quality measure (UIQM) [30] and the recently proposed CCF [29] metric, which is a weighted summation of the colorfulness, contrast, and fog density index, was performed on 1381 images from the test dataset provided in the EUVP dataset. The higher the value for each metric, the better the visual quality of the image. The quantitative results presented in Table III indicate that CRN-UIE does a better job compared to other GAN methods in terms of removing the effect of light attenuation and refraction, as well as improving the colorfulness in the enhanced underwater output images. FUnIE-GAN's outputs tend to have

[2][Online]. Available: http://irvlab.cs.umn.edu/resources/euvp-dataset

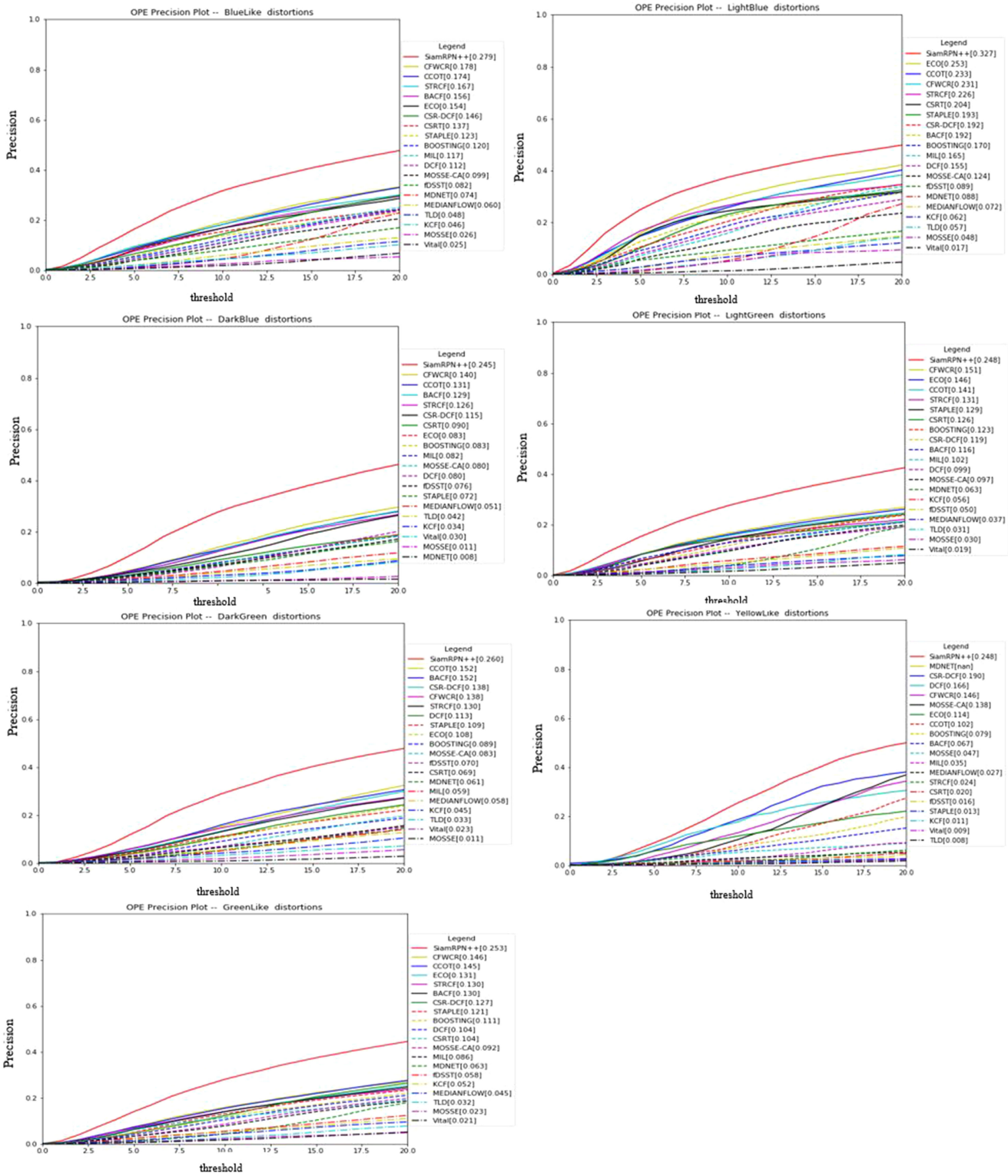


Fig. 10. OPE precision plots for various underwater visual data types. Trackers in legend are ranked from top to least performing using their AUC values.

better sharpness and better contrast as measured by the UISM and UIConM components of the UIQM metric. UGAN on the other hand has better overall UIQM score, whereas CRN-UIE has a much higher CCF score, which measures the quality of the underwater color image.

B. Improving Trackers Performance by Enhancing the Quality of Visual Data Using GAN

To improve tracking accuracy and precision of the trackers, we use an image enhancement approach to improve the quality

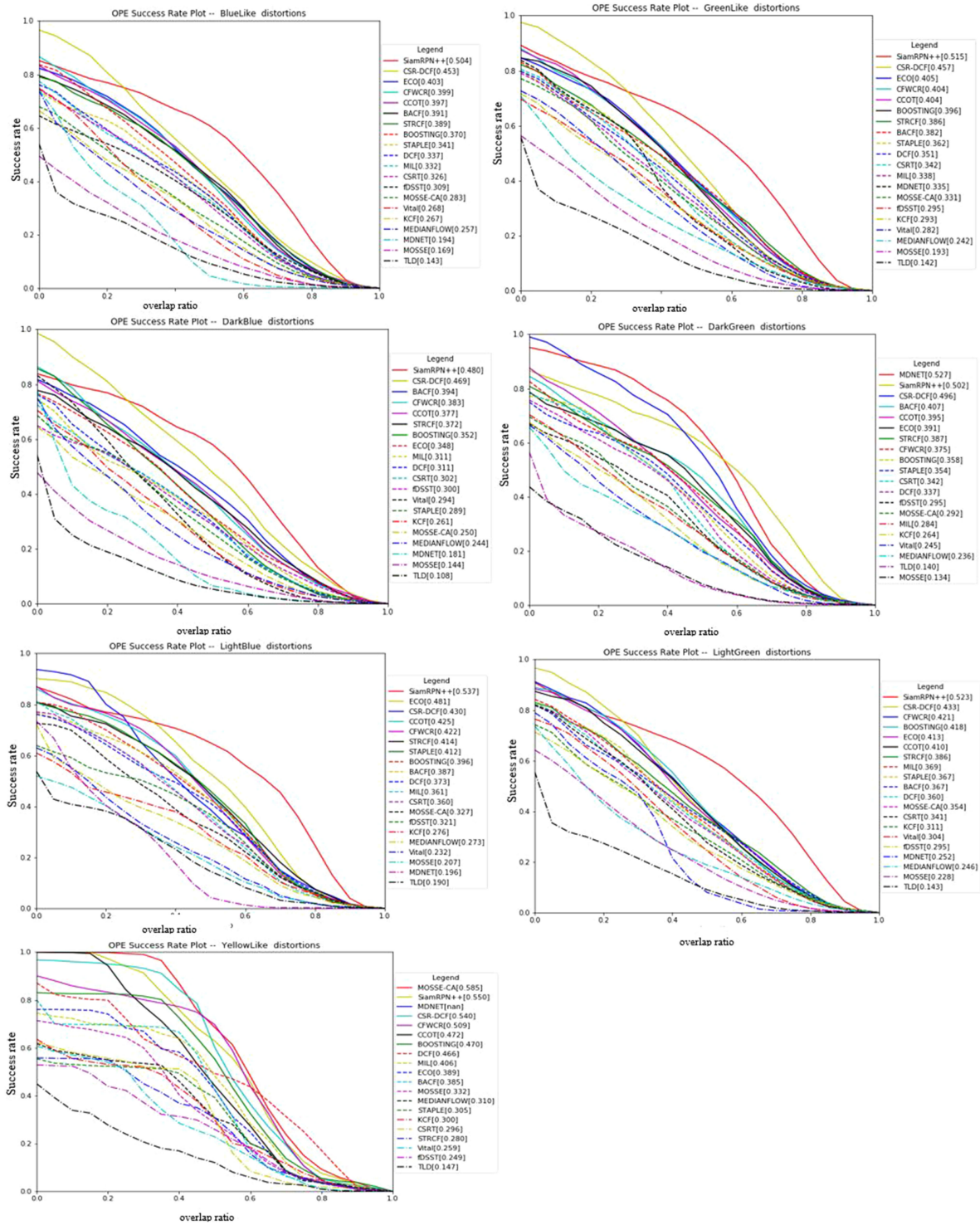


Fig. 11. OPE success rate plot for various types of underwater visual data. Trackers in legend are ranked from top to least performing using their AUC values.

of the underwater visual data by eliminating some inherent distortions. Given that trackers are found to have better performance in open-air environments than in underwater, we synthesize clear underwater images using the proposed CRN-UIE. We enhanced the UOT100 dataset to generate UOT100_enhanced and benchmark the performance of the same trackers on the enhanced visual data.

The OPE precision and success rate plots for the enhanced dataset are shown in Fig. 8. It can be seen that the proposed CRN-UIE GAN model can be used to enhance the performance of tracking algorithms on underwater visual data. A significant and consistent improvement in the precision as well as success rate of the selected trackers on the enhanced visual data can be observed. Fig. 8 compares benchmark performance on the entire

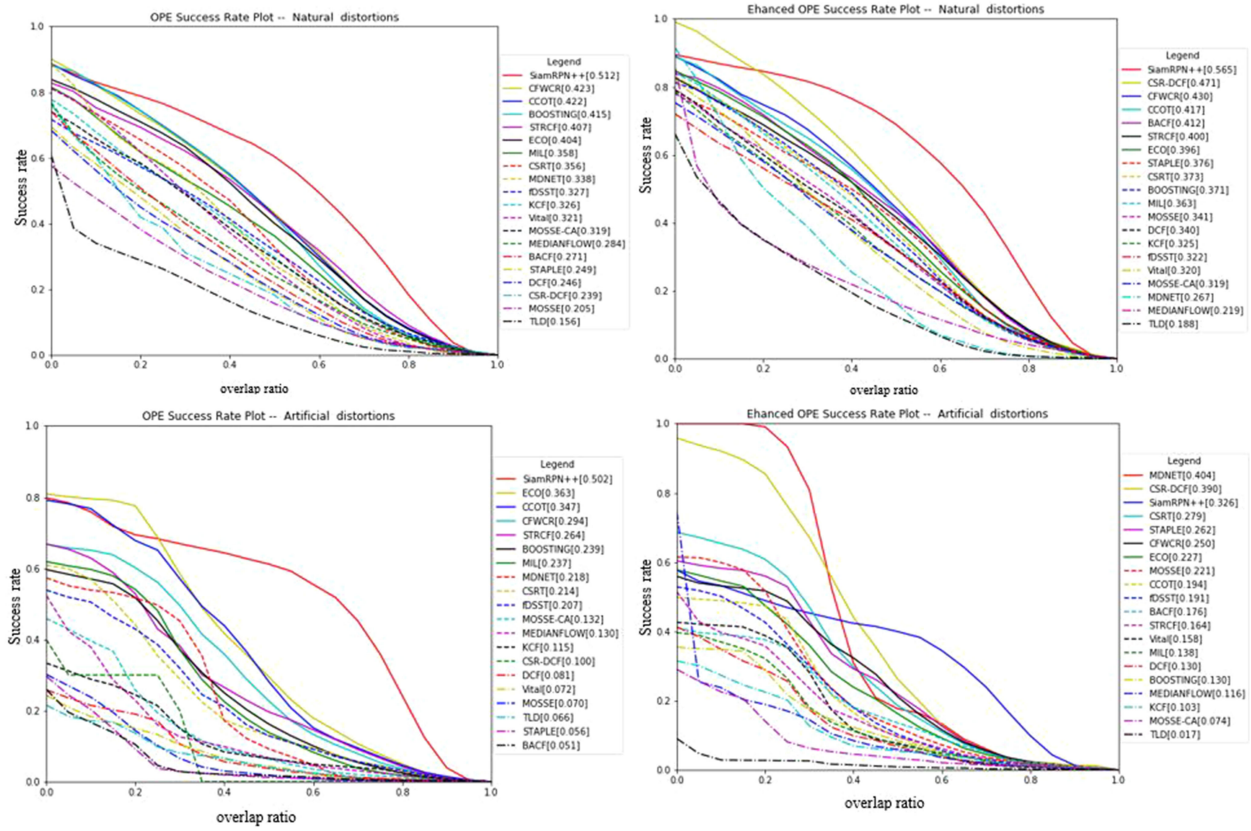


Fig. 12. Comparing the success rate benchmark plots on original (first row) versus enhanced (second row) data. Trackers in legend are ranked from top to least performing using their AUC values.

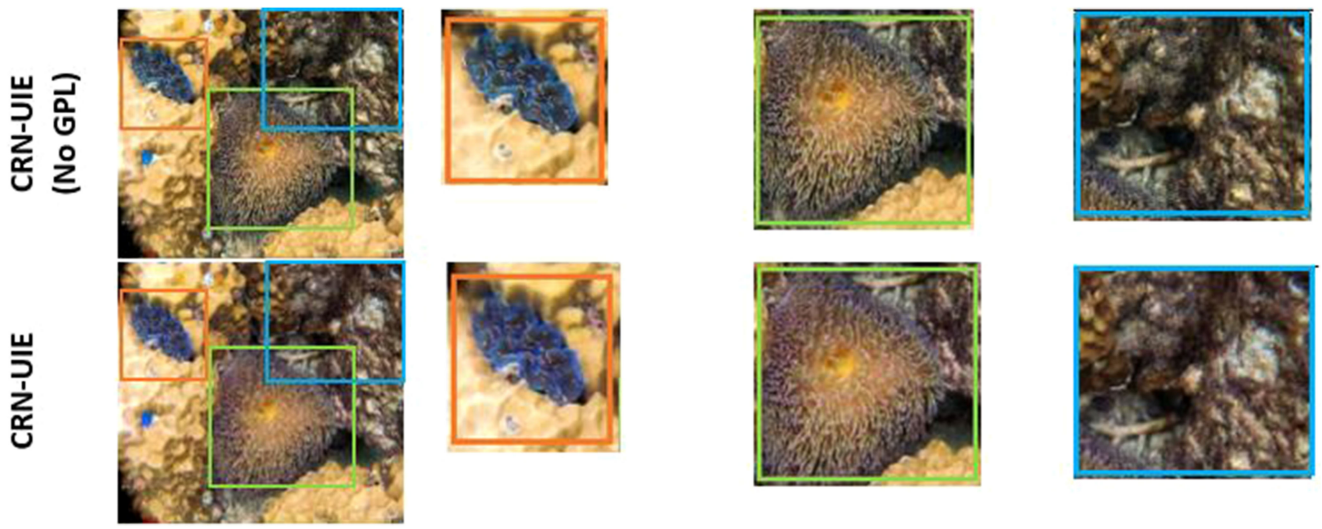


Fig. 13. Analyzing the impact of GP loss on the overall CRN-UIE objective function. The top row shows output result using CRN-UIE without gpl, whereas the second row is the final CRN-UIE with gpl. Zoom-ins at multiple regions show that the spl loss helps boost the color correction in the output image without saturation.

enhanced versus original dataset, whereas Fig. 9 compares the benchmark performance on natural versus artificial underwater data.

We observe consistent and significant increase in the accuracy of all trackers on the enhanced dataset as corroborated by the quantitative AUC results in Table IV.

Further analysis is presented in Appendixes A and C. In Appendix A, we provide performance analysis plots for different types of underwater distortions, and in Appendix C, we present tracking results and benchmark plots on data enhanced using the FUnIE-GAN and UGAN methods. We show that CRN-UIE outperforms these other state-of-the-art methods not only on

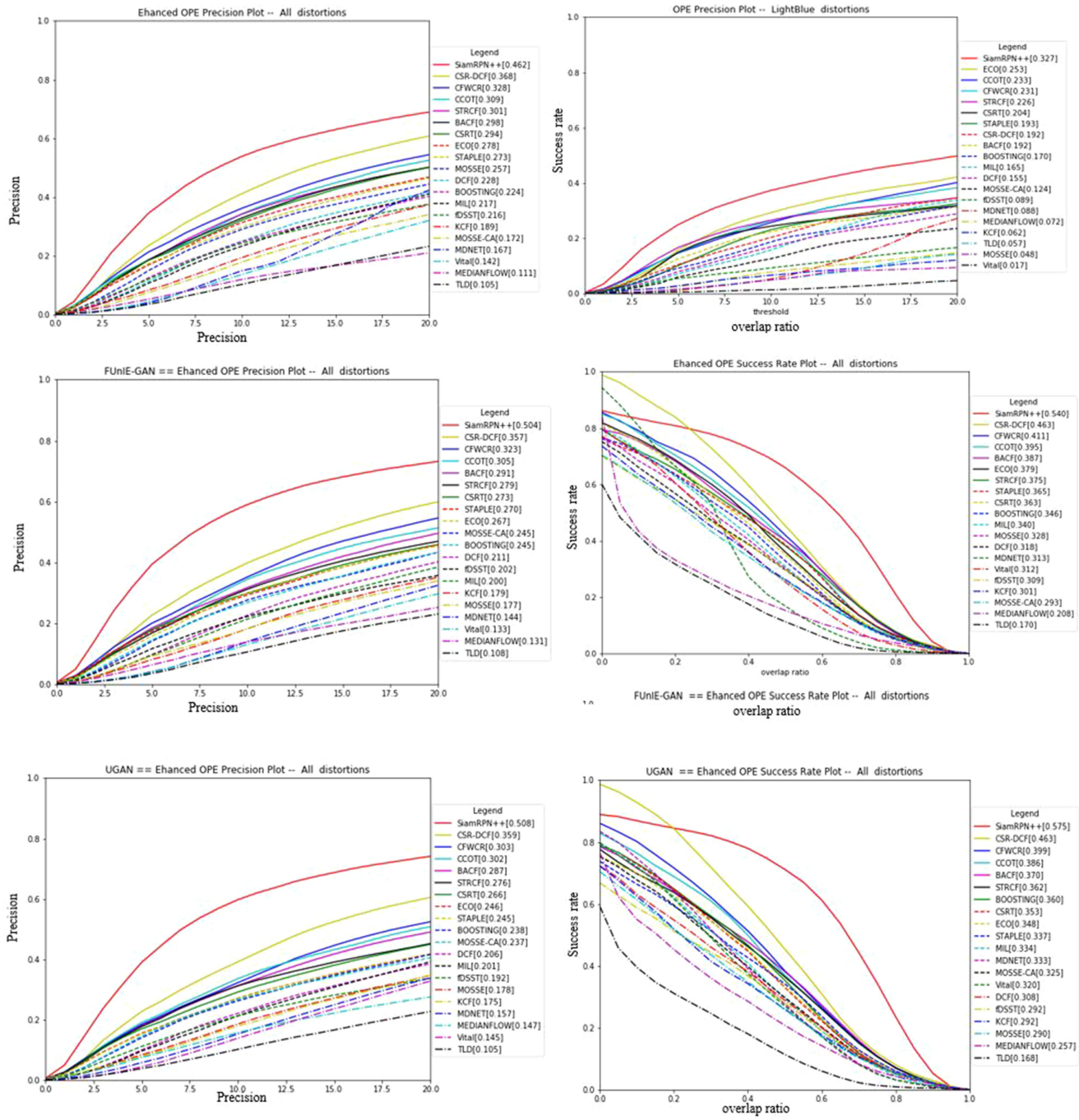


Fig. 14. Comparative benchmark results – Precision and Success rate plots on data enhanced using the proposed CRN-UIE (first row), FUnIE-GAN (second row), and UGAN (third row). Each row shows the precision and success rate plots for the corresponding enhancement method. It can be seen from the benchmark plots that most trackers perform better on underwater data enhanced using the prosed CRN-UIE enhancement method. SiamRPN++, however, tends to do better on data enhanced with both FUnIE-GAN and UGAN.

the image quality enhancement task, but also on the tracking performance enhancement of benchmarked trackers.

VI. Conclusion

In this article, the performance of state-of-the-art object tracking algorithms was shown to degrade considerably when tested on underwater environments as opposed to the open-air environments, due to the inherent distortions that affect the quality of underwater visual data. To help address this problem, we created a comprehensive underwater object tracking and benchmarking dataset to foster development of dedicated trackers, well suited for underwater environments and robust to the various inherent distortions. We further propose a new improved GAN model, i.e., the CRN-UIE model, for underwater image enhancement. Subsequent analysis shows that correcting the underwater distortions by translating the visual data to an enhanced/clear domain using our model significantly improves tracking accuracy in

underwater environments. The work done in this article will benefit sophistication of applications, such as underwater search and rescue operations, homeland and maritime security, deep ocean exploration, underwater robot navigation, and sea life monitoring. For the future work, we intend to explore better tracking evaluation metrics than OPE for underwater data. Additionally, multiple object tracking (MOT) will be considered.

APPENDIX A

We present here further performance analysis plots of the trackers on different types of distortions. Figs. 10 and 11 show OPE precision and Success Rate plots for various types of underwater visual data, respectively. Fig. 12 compares the success rate plots between trackers on original versus enhanced UOT100 dataset.

APPENDIX B

It is also necessary to emphasize the importance of the GP loss on the proposed CRN-UIE objective function. We realized during our experiments that adding the GP loss helped control the color correction in the output images. The GP loss helps preserve other high-frequency components, such as texture and tone. This is especially useful because colors get distorted in underwater environments. Fig. 13 shows sample output of our CRN-UIE architecture model with and without the GP loss.

APPENDIX C

Here, we present the benchmark tracking results on the enhanced UOT100 dataset using FUnIE-GAN and UGAN. It can be seen from the plots in Fig. 14 that enhancing the quality of the underwater visual data using these methods also improves the precision and success rate of trackers in underwater environments. However, these results also demonstrate that most trackers perform slightly better on UOT100_enhanced using the proposed CRN-UIE enhancement method, than with FUnIE-GAN and UGAN. SiamRPN++, VITAL, and TLD tend to perform slightly better on the data enhanced by both FUnIE-GAN and UGAN than CRN-UIE. We observe this trend in benchmark plots for all distortion types. It is equally important to point out that the performance ranking of the benchmarked trackers remains almost consistent across the various methods.

REFERENCES

[1] Y. Wang, W. Song, G. Fortino, L.-Z. Qi, W. Zhang, and A. Liotta, "An experimental-based review of image enhancement and image restoration methods for underwater imaging," *IEEE Access*, vol. 7, pp. 140233–140251, 2019.

[2] D. Mallet and D. Pelletier, "Underwater video techniques for observing coastal marine biodiversity: A review of sixty years of publications (1952-2012)," *Fisheries Res.*, vol. 154, pp. 44–62, 2014.

[3] M. Han, Z. Lyu, T. Qiu, and M. Xu, "A review on intelligence dehazing and color restoration for underwater images," *IEEE Trans. Syst., Man, Cybern., Syst.*, vol. 50, no. 5, pp. 1820–1832, May 2020.

[4] S. Corchs and R. Schettini, "Underwater image processing: State of the art of restoration and image enhancement methods," *EURASIP J. Adv. Signal Process.*, vol. 2010, 2010, Art. no. 746052.

[5] Y. Wu, J. Lim, and M. H. Yang, "Object tracking benchmark," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 37, no. 9, pp. 1834–1848, Sep. 2015.

[6] Y. Wu, J. Lim, and M.-H. Yang, "Online object tracking: A benchmark," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2013, pp. 2411–2418.

[7] "VOT challenge| challenges." Accessed: Sep. 30, 2019. [Online]. Available: http://www.votchallenge.net/challenges.html

[8] A. Milan, L. Leal-Taixe, I. Reid, S. Roth, and K. Schindler, "MOT16: A benchmark for multi-object tracking," Accessed: Sep. 30, 2019. Mar. 2016. [Online]. Available: http://arxiv.org/abs/1603.00831

[9] L. Leal-Taixé, A. Milan, I. Reid, S. Roth, and K. Schindler, "MOTChallenge 2015: Towards a benchmark for multi-target tracking," Apr. 2015. Accessed: Sep. 30, 2019. [Online]. Available: http://arxiv.org/abs/1504.01942

[10] J. F. Henriques, R. Caseiro, P. Martins, and J. Batista, "High-speed tracking with kernelized correlation filters," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 37, no. 3, pp. 583–596, Mar. 2015.

[11] M. Danelljan, G. Bhat, F. S. Khan, and M. Felsberg, "ECO: Efficient convolution operators for tracking," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 6931–6939.

[12] L. Bertinetto, J. Valmadre, J. F. Henriques, A. Vedaldi, and P. H. S. Torr, "Fully-convolutional Siamese networks for object tracking," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2016, pp. 850–865.

[13] Y. Li and J. Zhu, "A scale adaptive kernel correlation filter tracker with feature integration," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2014, pp. 254–265.

[14] H. K. Galoogahi, A. Fagg, and S. Lucey, "Learning background-aware correlation filters for visual tracking," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2017, pp. 1144–1152.

[15] F. Li, C. Tian, W. Zuo, L. Zhang, and M.-H. Yang, "Learning spatial-temporal regularized correlation filters for visual tracking," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 4904–4913.

[16] L. Bertinetto, J. Valmadre, S. Golodetz, O. Miksik, and P. H. S. Torr, "Staple: Complementary learners for real-time tracking," in *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 1401–1409.

[17] L. Kezebou, V. Oludare, K. Panetta, and S. S. Agaian, "Underwater object tracking benchmark and dataset," in *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recognit.*, Nov. 2019, pp. 1–6.

[18] D. Berman, D. Levy, S. Avidan, and T. Treibitz, "Underwater single image color restoration using haze-lines and a new quantitative dataset," *IEEE Trans. Pattern Anal. Mach. Intell.*, early access, 2020, doi: 10.1109/TPAMI.2020.2977624.

[19] A. S. Abdul Ghani and N. A. Mat Isa, "Enhancement of low quality underwater image through integrated global and local contrast correction," *Appl. Soft Comput. J.*, vol. 37, pp. 332–344, Dec. 2015.

[20] K. Panetta, C. Gao, and S. Agaian, "No reference color image contrast and quality measures," *IEEE Trans. Consum. Electron.*, vol. 59, no. 3, pp. 643–651, Aug. 2013.

[21] S. S. Agaian, B. Silver, and K. A. Panetta, "Transform coefficient histogram-based image enhancement algorithms using contrast entropy," *IEEE Trans. Image Process.*, vol. 16, no. 3, pp. 741–758, Mar. 2007.

[22] S. S. Agaian, K. Panetta, and A. M. Grigoryan, "Transform-based image enhancement algorithms with performance measure," *IEEE Trans. Image Process.*, vol. 10, no. 3, pp. 367–382, Mar. 2001.

[23] A. J. Chrispin and R. Nagaraj, "Deblurring underwater image degradations based on adaptive regularization," in *Proc. IEEE Int. Conf. Comput. Intell. Comput. Res.*, Dec. 2018, pp. 1–7.

[24] C. Fabbri, M. J. Islam, and J. Sattar, "Enhancing underwater imagery using generative adversarial networks," in *Proc. IEEE Int. Conf. Robot. Autom.*, Sep. 2018, pp. 7159–7165.

[25] M. Liu, F. Yuan, Y. Zhu, and E. Cheng, "Generating underwater images by GANs and similarity measurement," in *Proc. OCEANS - MTS/IEEE Kobe Techno-Oceans*, May 2018, pp. 1–5.

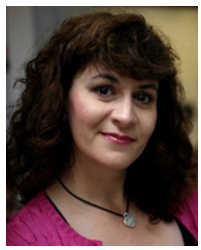
[26] Y. Hashisho, M. Albadawi, T. Krause, and U. F. von Lukas, "Underwater color restoration using U-Net denoising autoencoder," in *Proc. 11th Int. Symp. Image Signal Process. Anal.*, 2019, doi: 10.1109/ISPA.2019.8868679

[27] X. Yu, Y. Qu, and M. Hong, "Underwater-GAN: Underwater image restoration via conditional generative adversarial network," in *Proc. Pattern Recognit. Inf. Forensics (ICPR)*. vol. 11188, 2019, pp. 66–75.

[28] Y. Guo, H. Li, and P. Zhuang, "Underwater image enhancement using a multiscale dense generative adversarial network," *IEEE J. Ocean. Eng.*, vol. 45, no. 3, pp. 862–870, Jul. 2020.

[29] Y. Wang *et al.*, "An imaging-inspired no-reference underwater color image quality assessment metric," *Comput. Elect. Eng.*, vol. 70, pp. 904–913, Aug. 2018.

[30] K. Panetta, C. Gao, and S. Agaian, "Human-visual-system-inspired underwater image quality measures," *IEEE J. Ocean. Eng.*, vol. 41, no. 3, pp. 541–551, Jul. 2016.
[31] "OTB toolkit | OTB-toolkit." Accessed: Oct. 1, 2019. [Online]. Available: https://zhouyzzz.github.io/otb-toolkit/
[32] S. Bei, Z. Zhen, L. Wusheng, D. Liebo, and L. Qin, "Visual object tracking challenges revisited: VOT vs. OTB," *PLoS One*, vol. 13, no. 9, Sep. 2018, Art. no. e0203188.
[33] M. Kristan *et al.*, "A novel performance evaluation methodology for single-target trackers," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 38, no. 11, pp. 2137–2155, Nov. 2016.
[34] "MOT challenge - Results." Accessed: Aug. 14, 2020. [Online]. Available: https://motchallenge.net/results/MOT17?det=Private
[35] Y. Wang, X. Luo, L. Luo, H. Zhang, and X. Wei, "UAV tracking based on saliency detection," *Soft Comput.*, vol. 24, no. 16, pp. 12149–12162, Jan. 2020.
[36] A. Li, M. Lin, Y. Wu, M. H. Yang, and S. Yan, "NUS-PRO: A new visual tracking challenge," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 38, no. 2, pp. 335–349, Feb. 2016.
[37] A. Moudgil and V. Gandhi, "Long-term visual object tracking benchmark," in *Proc. Asian Conf. Comput. Vis. (ACCV)*, vol. 11362, Dec. 2017, pp. 629–645.
[38] "Long-term visual object tracking benchmark." Accessed: Aug. 14, 2020. [Online]. Available: https://amoudgl.github.io/tlp/
[39] "Long-term tracking." Accessed: Aug. 14, 2020. [Online]. Available: https://oxuva.github.io/long-term-tracking-benchmark/
[40] J. Valmadre *et al.*, "Long-term tracking in the wild: A benchmark," in *Proc. Euo. Conf. Comput. Vis. (ECCV)*, 2018, pp. 670–685.
[41] J. Xiao, R. Stolkin, and A. Leonardis, "Single target tracking using adaptive clustered decision trees and dynamic multi-level appearance models," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2015, pp. 4978–4987.
[42] B. Babenko, M.-H. Yang, and S. Belongie, "Visual tracking with online multiple instance learning," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2009, pp. 983–990.
[43] W. Zhong, H. Lu, and M. H. Yang, "Robust object tracking via sparsity-based collaborative model," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, 2012, pp. 1838–1845.
[44] V. Ramalakshmi and M. G. Alex, "Visual object tracking using discriminative correlation filter," in *Proc. Int. Conf. Commun. Electron. Syst.*, 2016, doi: 10.1109/CESYS.2016.7889887.
[45] M. Danelljan, G. Hager, F. S. Khan, and M. Felsberg, "Discriminative scale space tracking," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 8, pp. 1561–1575, Aug. 2017.
[46] M. Danelljan, G. Häger, S. Khan, and M. Felsberg, "Learning spatially regularized correlation filters for visual tracking," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2015, pp. 4310–4318.
[47] C. Ma, J. Huang, X. Yang, and M. Yang, "Hierarchical convolutional features for visual tracking," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2015, pp. 3074–3082.
[48] M. Danelljan, A. Robinson, F. S. Khan, and M. Felsberg, "Beyond correlation filters: Learning continuous convolution operators for visual tracking," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, vol. 9909, 2016, pp. 472–488.
[49] H. Nam and B. Han, "Learning multi-domain convolutional neural networks for visual tracking," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2016, pp. 4293–4302.
[50] S. Hare *et al.*, "Struck: Structured output tracking with kernels," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 38, no. 10, pp. 2096–2109, Oct. 2016.
[51] J. Valmadre, L. Bertinetto, J. Henriques, A. Vedaldi, and P. H. S. Torr, "End-to-end representation learning for correlation filter based tracking," in *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 5000–5008.
[52] A. Lukežič, T. Vojíř, L. Čehovin Zajc, J. Matas, and M. Kristan, "Discriminative correlation filter tracker with channel and spatial reliability," *Int. J. Comput. Vis.*, vol. 126, no. 7, pp. 671–688, Jul. 2018.
[53] M. Mirza and S. Osindero, "Conditional generative adversarial nets." Accessed: Nov. 9, 2019. Nov. 2014. [Online]. Available: http://arxiv.org/abs/1411.1784
[54] M. Y. Liu, T. Breuel, and J. Kautz, "Unsupervised image-to-image translation networks," *Adv. Neural Inf. Process. Syst.*, vol. 2017, pp. 701–709, 2017.
[55] P. Isola, J.-Y. Zhu, T. Zhou, and A. A. Efros, "Image-to-image translation with conditional adversarial networks," in *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 5967–5976.
[56] T.-C. Wang, M.-Y. Liu, J.-Y. Zhu, A. Tao, J. Kautz, and B. Catanzaro, "High-Resolution image synthesis and semantic manipulation with conditional GANs," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 8798–8807.
[57] J.-Y. Zhu, T. Park, P. Isola, and A. A. Efros, "Unpaired image-to-image translation using cycle-consistent adversarial networks," in *Proc. Conf. IEEE Int. Conf. Comput. Vis.*, 2017, pp. 2223–2232.
[58] J. Li, K. A. Skinner, R. M. Eustice, and M. Johnson-Roberson, "WaterGAN: Unsupervised generative network to enable real-time color correction of monocular underwater images," *IEEE Robot. Autom. Lett.*, vol. 3, no. 1, pp. 387–394, Jan. 2018.
[59] M. J. Islam, Y. Xia, and J. Sattar, "Fast underwater image enhancement for improved visual perception," *IEEE Robot. Autom. Lett.*, vol. 5, no. 2, pp. 3227–3234, Apr. 2020.
[60] H. Grabner, M. Grabner, and H. Bischof, "Real-time tracking via on-line boosting," in *Proc. Brit. Mach. Vis. Conf.*, 2006, pp. 47–56.
[61] Z. He, Y. Fan, J. Zhuang, Y. Dong, and H. Bai, "Correlation filters with weighted convolution responses," in *Proc. IEEE Int. Conf. Comput. Vis. Workshops*, 2017, pp. 1992–2000.
[62] Z. Kalal, K. Mikolajczyk, and J. Matas, "Forward-backward error: Automatic detection of tracking failures," in *Proc. Int. Conf. Pattern Recognit.*, 2010, pp. 2756–2759.
[63] D. S. Bolme, J. R. Beveridge, B. A. Draper, and Y. M. Lui, "Visual object tracking using adaptive correlation filters," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, 2010, pp. 2544–2550.
[64] Q. Wang, L. Zhang, L. Bertinetto, W. Hu, and P. H. S. Torr, "Fast online object tracking and segmentation: A unifying approach," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 1328–1338.
[65] B. Li, W. Wu, Q. Wang, F. Zhang, J. Xing, and J. Yan, "SiamRPN++: Evolution of Siamese visual tracking with very deep networks," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 4277–4286.
[66] Z. Kalal, K. Mikolajczyk, and J. Matas, "Tracking-learning-detection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 34, no. 7, pp. 1409–1422, Jul. 2012.
[67] Y. Song *et al.*, "VITAL: Visual tracking via adversarial learning," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 8990–8999.
[68] I. J. Goodfellow *et al.*, "Generative adversarial networks," Accessed: Mar. 1, 2020. 2014. [Online]. Available: http://www.github.com/goodfeli/adversarial
[69] J. Johnson, A. Alahi, and L. Fei-Fei, "Perceptual losses for real-time style transfer and super-resolution," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, vol. 9906, 2016, pp. 694–711.
[70] M. S. Sarfraz, C. Seibold, H. Khalid, and R. Stiefelhagen, "Content and colour distillation for learning image translations with the spatial profile loss," Aug. 2019. Accessed: Mar. 18, 2020. [Online]. Available: http://arxiv.org/abs/1908.00274



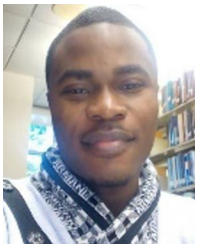
Karen Panetta (Fellow, IEEE) received the B.S. degree in computer engineering from Boston University, Boston, MA, USA, in 1985, and the M.S. and Ph.D. degrees in electrical engineering from Northeastern University, Boston, MA, USA, in 1987 and 1994, respectively.

She is currently a Dean of Graduate Engineering Education and a Professor with the Department of Electrical and Computer Engineering, Tufts University, Medford, MA, USA, and the Director of the Dr. Panetta's Vision and Sensing System Laboratory. Her research interests include developing efficient algorithms for simulation, modeling, signal, and image processing for biomedical and security applications.

Dr. Panetta is the President-Elect of the IEEE Eta Kappa Nu.

Landry Kezebou (Graduate Student Member, IEEE) received the Bachelor's of Engineering degree in electrical engineering from Ahmadu Bello University, Zaria, Nigeria, in 2014, and the M.S. degree in electrical and computer engineering from Tufts University, Medford, MA, USA, in 2019. He is currently working toward the Ph.D. degree with Tufts ECE Department.

His current research interest includes using the latest computer vision and deep learning technologies for highway intelligent transportation systems, as well as object detection and object tracking. He is also interested in underwater object tracking, image enhancement, and generative adversarial networks.



Victor Oludare (Graduate Student Member, IEEE) received the B.S degree in electrical engineering from the Federal University of Technology Minna, Minna, Nigeria, in 2014, and the M.S. degree from Tufts University, Medford, MA, USA, in 2018, where he is currently working toward the Ph.D. degree in electrical engineering.

His research interests include the use of computer vision techniques and deep learning architectures for conservation purposes, object detection, recognition and tracking, and image enhancement.

Sos Agaian (Fellow, IEEE) received the M.S. degree in mathematics and mechanics from Yerevan University, Yerevan, Armenia, in 1968, and the Ph.D. degree in mathematics and physics from the Steklov Institute of Mathematics, Russian Academy of Sciences, Moskva, Russia, in 1975, where he also received the Doctor of Engineering Sciences degree from the Institute of the Control System in 1985.

He is currently a Distinguished Professor with the City University of New York, New York, NY, USA. His research interests include computational vision and machine learning, multimodal data fusion, signal/image processing modeling, multimodal biometric and digital forensics, three-dimensional imaging sensors, information processing and security, and biomedical and health informatics.

Dr. Agaian is a Fellow of the Society for Imaging Science & Technology (SPIE) and the American Association for the Advancement of Science.