SeaScan: An Energy-Efficient Underwater Camera for Wireless 3D Color Imaging

Nazish Naeem
nazishn@mit.edu
Massachusetts Institute of Technology
Cambridge, Massachusetts, USA

Jack Rademacher
jradema@mit.edu
Massachusetts Institute of Technology
Cambridge, Massachusetts, USA

Ritik Patnaik
rik01@mit.edu
Massachusetts Institute of Technology
Cambridge, Massachusetts, USA

Tara Boroushaki
tarab@mit.edu
Massachusetts Institute of Technology
Cambridge, Massachusetts, USA

Fadel Adib
fadel@mit.edu
Massachusetts Institute of Technology
Cambridge, Massachusetts, USA

ABSTRACT

We present the design, implementation, and evaluation of SeaScan, an energy-efficient camera for 3D imaging of underwater environments. At the core of SeaScan's design is a trinocular lensing system, which employs three ultra-low-power monochromatic image sensors to reconstruct color images. Each of the sensors is equipped with a different filter (red, green, and blue) for color capture. The design introduces multiple innovations to enable reconstructing 3D color images from the captured monochromatic ones. This includes an ML-based cross-color alignment architecture to combine the monochromatic images. It also includes a cross-refractive compensation technique that overcomes the distortion of the wide-angle imaging of the low-power CMOS sensors in underwater environments. We built an end-to-end prototype of SeaScan, including color filter integration, 3D reconstruction, compression, and underwater backscatter communication. Our evaluation in real-world underwater environments demonstrates that SeaScan can capture underwater color images with as little as 23.6 mJ, which represents 37× reduction in energy consumption in comparison to the lowest-energy state-of-the-art underwater imaging system. We also report qualitative and quantitative evaluation of SeaScan's color reconstruction and demonstrate its success in comparison to multiple potential alternative techniques (both geometric and ML-based) in the literature. SeaScan's ability to image underwater environments at such low energy opens up important applications in long-term monitoring for ocean climate change, seafood production, and scientific discovery.

ACM MobiCom '24, November 18–22, 2024, Washington D.C., DC, USA

ACM ISBN 979-8-4007-0489-5/24/11
https://doi.org/10.1145/3636534.3690661

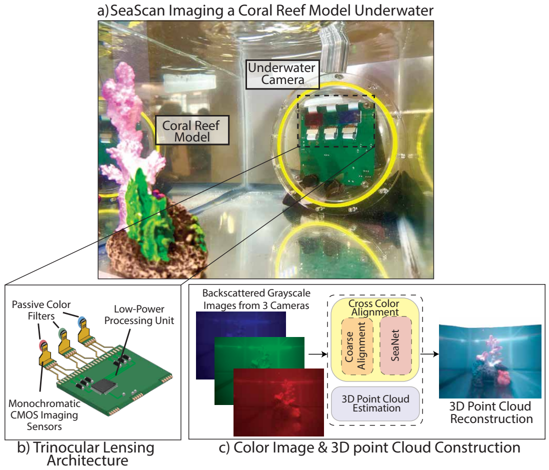

a) SeaScan Imaging a Coral Reef Model Underwater
Underwater Camera
Coral Reef Model
Passive Color Filters
Low-Power Processing Unit
Monochromatic CMOS Imaging Sensors
Backscattered Grayscale Images from 3 Cameras
Cross Color Alignment
Coarse Alignment
SeaNet
3D Point Cloud Estimation
3D Point Cloud Reconstruction
b) Trinocular Lensing Architecture
c) Color Image & 3D point Cloud Construction

Figure 1: System Overview. (a) shows SeaScan imaging a coral reef model. (b) depicts our trinocular lensing system. (c) shows the processing pipeline.

CCS CONCEPTS

• **Networks** → *Sensor networks*; • **Computing methodologies** → **Machine learning**.

KEYWORDS

Subsea IoT, Energy Efficient Imaging, Deep Learning, Backscatter

ACM Reference Format:
Nazish Naeem, Jack Rademacher, Ritik Patnaik, Tara Boroushaki, and Fadel Adib. 2024. SeaScan: An Energy-Efficient Underwater Camera for Wireless 3D Color Imaging. In *The 30th Annual International Conference on Mobile Computing and Networking (ACM MobiCom '24), November 18–22, 2024, Washington D.C., DC, USA.* ACM, New York, NY, USA, 15 pages. https://doi.org/10.1145/3636534.3690661

1 INTRODUCTION

Long-term underwater imaging of marine animals, plants and the ocean is critical to discovering new species, monitoring the impact of human activities on the ocean, and sustainably monitoring and maintaining aquaculture farms [25, 27, 40]. For example, underwater cameras can help aquaculture farms in regulating feeding patterns, detecting diseases, and monitoring both contamination in their food chains and waste outflow to their surrounding environments [17, 63]. Moreover, long-term underwater imaging can play an instrumental role in oceanography and climatology, allowing us to study and understand sub-sea geological behavior such as submarine volcanoes and hydrothermal vents, nutrient transport dynamics of the ocean carbon cycle, and habitat deterioration such as coral reef bleaching and acidification [16, 31, 44].

Recent advances in underwater imaging have taken important steps toward this vision through the introduction of underwater cameras that consume joule-level energies per image capture [2]. Yet, it remains desirable to further reduce the energy consumption of these underwater cameras to extend their battery lifetime (proportionately). For example, if we can design underwater cameras to run on sub-joules or tens of milli-joules, then they could last for many years on coin cell batteries,[1] paving the way for long-term monitoring in challenging and remote underwater environments.

In this paper, we ask whether we can push the boundaries of underwater color imaging by further reducing the energy consumption of underwater cameras. One of the key challenges in doing so lies in how state-of-the-art imaging systems work. Specifically, these systems use ultra-low-power monochromatic CMOS image sensors, but reconstruct color images by sequentially emitting flashes of red, green, and blue light using LEDs; capturing a monochrome image with every flash; then, linearly combining these images to form a color image. Since the sequential flashing and buffering consume more than 80% of the energy during the image capture phase alone, eliminating the need for them would significantly reduce the energy consumption of the camera. A natural alternative is to use a color imaging sensor; however, these sensors typically consume at least an order of magnitude more energy [24, 26, 45, 46] than commercial-off-the-shelf (COTS) monochromatic CMOS imaging sensors [23].

We introduce SeaScan, an energy-efficient underwater imaging system for wireless 3D color imaging. SeaScan leverages COTS ultra-low power monochromatic imaging sensors to capture color images without flashing LEDs. Not only can SeaScan capture rich color images, but it can also construct 3D point clouds and wirelessly transmit images at ultra-low-energy via underwater backscatter.

At the core of SeaScan is a trinocular lensing design that is inspired by how human eyes (as well as color cameras) work. As depicted in Fig. 1(b), SeaScan consists of three monochromatic CMOS lenses equipped with different passive color (red, green, and blue) filters. In principle, the images captured from these filter-equipped lenses can be combined to form a color (RGB) image. However, translating this design into a practical system faces several challenges. In typical color cameras, adjacent photodetectors capture different colors of the same part of a scene, and can be linearly combined to reconstruct a color image (The same happens with different cone cells in the human eye). In contrast, SeaScan captures different colors from spatially separated lenses. As a result, the same object appears at different locations (pixels) in three resulting images. Thus, a direct combination of the images would result in poor color reconstruction, as shown in Fig. 2(d). One might assume that knowing the position of the three lenses and applying a simple geometric transformation would enable us to combine the three images for color reconstruction. But, the translational and rotational relationship of the position of the cameras does not directly translate into the captured images, thus resulting in poor color reconstruction as shown in Fig. 2(e).

To overcome this challenge, SeaScan introduces *Cross Color Alignment*, a learning-based approach for reconstructing color images from its trinocular lensing design. One thing to note here is that the color image reconstruction is not entirely an image alignment problem but also an inference problem. This is because we are capturing images from three spatially separated cameras. As a result, it is possible that a part of the scene is visible in one camera but occluded from another camera's field of view. Therefore, Cross Color Alignment not only aligns but also infers the missing colors from the contextual information. Our method breaks the color image reconstruction problem into two steps. First, we coarsely align the images using a RANSAC-flow architecture [50]. Second, we feed the aligned images into to *SeaNet*, a custom-designed 2D convolutional neural network, that performs fine alignment and color inference on the misaligned and missing parts of the images.

Another challenge in enabling color imaging with SeaScan arises from the low-power COTS CMOS imaging sensor's built-in wide-angle lens. This wide-angle lens is useful for many applications including surveillance, vehicle cameras, or multi-object detection, but introduces distortion to the images [67] that adds new complexities to the image alignment process. Since SeaScan relies on a learning-based approach to reconstruct color images, the presence of such wide-angled distortions in the images significantly impacts the performance of our algorithm. Moreover, filtering the incoming light from the scene into the appropriate red, green, and blue channels introduces additional challenges with reflections and artifacts in the resulting images. These artifacts not only negatively impact image and color alignment, but manifest new features in the captured images that cause the depth estimation process to incorrectly estimate the three dimensional characteristics of the scene.

[1] A typical coin cell battery of 230mAh at 2V. Thus, a 50 mJ-camera that captures an image of corals once per hour can last for about 4 years.

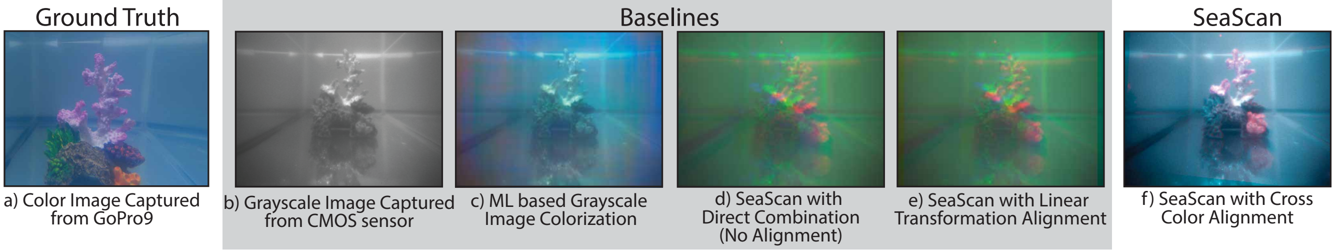

Ground Truth
Baselines
SeaScan
a) Color Image Captured from GoPro9
b) Grayscale Image Captured from CMOS sensor
c) ML based Grayscale Image Colorization
d) SeaScan with Direct Combination (No Alignment)
e) SeaScan with Linear Transformation Alignment
f) SeaScan with Cross Color Alignment

Figure 2: This figure shows (a) Image captured from a GoPro camera. (b) Grayscale image of the same scene captured from a low power CMOS imaging sensor. (c) Output image of a machine learning based colorization method when image in part (b) is given as input. (d) Image captured by SeaScan if the three images were directly combined without any alignment. (e) Image captured by SeaScan if the three images were linearly transformed before combining. f)Image captured by SeaScan if the three images combined using the Cross Color Alignment method described in the paper.

To overcome these challenges, SeaScan exploits the physical characteristics of light propagation at medium boundaries. Specifically, we design the waterproofing encapsulation of SeaScan such that the refraction across the water-air boundary inverts the impact of distortion caused by our imaging sensor's wide-angle lens. By carefully selecting the water-air boundary layer, we can constrain the field of view of the camera and capture rectilinear images of the underwater scene that can be fed directly to our color alignment and depth estimation algorithms. Furthermore, instead of using highly selective and expensive optical filters for the red, green, and blue channels, SeaScan leverages cheaper and more widely available gel filters. These filters not only maintain the wavelength-selective properties needed for color estimation but significantly improve reflection artifacts that arise from the interaction of light between the scene, our waterproof encapsulation, the filters, and the image sensors.

We implemented an end-to-end prototype of SeaScan shown in Fig. 1(a) and tested it across multiple kinds of images and baseline metrics. We extend our core colorization in three ways to push the boundaries of underwater imaging, 1) SeaScan extends the capability of the imaging system to 3D by leveraging state-of-the-art image-based depth estimation algorithms. This allows SeaScan to recreate the 3D point cloud of the captured environments and gives a deeper understanding of the underwater world. 2) We employ underwater backscatter for wireless communication of the underwater images at extremely low power. 3) We further incorporate image compression to reduce the data that needs to be transmitted and therefore reduce the communication time. As a result, we further reduce the energy consumption of SeaScan compared to the state-of-the-art.

We evaluated SeaScan in more than 80 real-world underwater experiments, and compared it to multiple baselines including a state-of-the-art NN-based Colorization, Direct Combination, and Linear Transformation Alignment:

- For the image capture phase SeaScan consumes $23.6mJ$ of energy in comparison to the state-of-the-art [2] that consumes $894.2mJ$ of energy for color imaging, demonstrating more than 37× reduction in the energy consumption.
- We compare Cross Color Alignment with two types of baselines 1) learning-based grayscale colorization method and 2) alignment-based methods. Our method shows a 50% improvement in the CIEDE2000 in comparison to the best baseline on the test dataset.
- We also show the effectiveness of SeaScan in tasks like KNN-based color classification on images captured from SeaScan. SeaScan is 40% more accurate than a random guess in correctly identifying the colors of a captured object, and 20% more accurate than the closest baselines.

Contributions SeaScan is the most energy-efficient underwater color imaging system, and the first camera capable of 3D color imaging in underwater environments. Its design introduces multiple innovations including a trinocular lensing design, learning-based color alignment method, and cross-refractive design to capture color images at ultra-low energy. The paper also contributes an end-to-end prototype implementation and evaluation of the system, with integrated compression and underwater backscatter communication.

2 TRINOCULAR PASSIVE IMAGING

2.1 The Alignment Problem

To understand why color reconstruction from a trinocular lensing system is challenging, consider Fig. 3a, where we depict the camera capturing a color image of a simplified scene containing two simple cubes: one red and one blue. Since the three monochromatic sensors are equipped with passive color filters, they not only see different viewpoints of the same scene but also the same object may exhibit disparate features across the three images (since many features are color-based).[2]

Fig. 3b shows the images captured by the trinocular lensing system. Note that the three sensors capture grayscale images; however, for simplicity, we represent images captured by red, green, and blue color filters as red, green, and blue. Note that both cubes appear at different locations across the three images. Moreover, note that the distance between the pixels of the two cubes is not the same in the three images. This is

[2] Note that because each monochromatic sensor has a form factor of 1.2x2 cm, they cannot be perfectly co-located, which causes target displacement in the captured images.

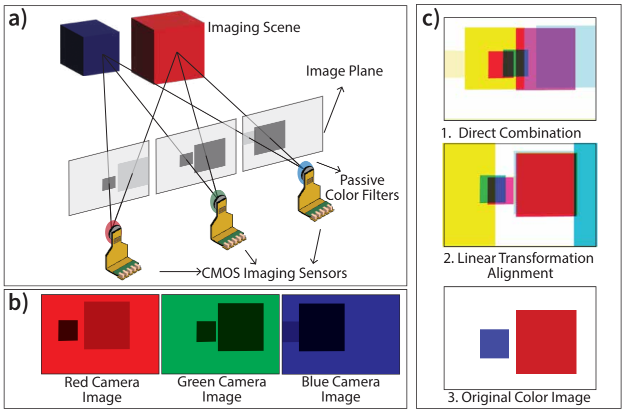


Figure 3: Trinocular Lensing System. a) shows the three CMOS imaging sensors equipped with passive color filters imaging two cubes. The camera viewpoint can be seen on the imaging plane. b) shows the images captured from the three monochrome cameras equipped with color filters. c) shows the results of 1. Direct Combination, 2. Linear Transformation alignment and 3. Original Color Image if captured from a color camera

because the position of an object in the image depends on the relative position of the camera from the object.

Now that we have three images from the three cameras the next step is to form a color image using the three images. A naive solution is to simply combine the three images captured from the trinocular lensing system to form a color image as shown in image 1 in Fig. 3c. However, since the cameras are looking at the object from distinct physical positions, the objects appear at different locations in all three images, and therefore, the final color image results in distorted colors.

Another potential solution is to consider the fixed camera positions and determine transformations between the camera images, anticipating an inverse transformation to align the images correctly before incorporating them into the color image channels. While this seems reasonable, it's important to note that camera transformations don't directly translate into image transformations [55]. Image 2 in Fig. 3c shows the result of aligning the images using the transformation in the position of the cameras. In the rest of the paper, we refer to this type of alignment as linear transformation alignment.

2.2 Trinocular Cross Color Alignment

To overcome the above challenges and capture color images from the trinocular lensing system, SeaScan introduces *Cross Color Alignment*, a learning-based approach for aligning the monochrome images captured from the three filtered monochromatic sensors. This method operates in two stages:.

Coarse Alignment: In the past, researchers have extensively looked into the problem of image alignment to enable tasks like visual localization, texture transfer, predicting the 2D geometry optical flow estimation, etc [48]. However, these methods alone are not sufficient for solving the problem in our trinocular lensing system, where the three images of interest represent different channels.

To address this problem, SeaScan takes a multi-stage approach, and leverages these past approaches to perform the first stage of alignment.

In the coarse alignment stage, SeaScan adapts RANSAC-flow net [51] as a backbone to align the three images in different color channels. It does so in a pairwise fashion. Specifically, we chose one of the captured images (say green camera image[3]) and used it as the reference/target image as shown in Fig. 4. The other two images are aligned with this target image in two steps. In the first step the red camera image is coarsely aligned with the green image and the result is combined with the green image to form a color image that only has a red and green channel (the blue channel is all zero) shown in Fig. 4 as red-green image. Now in the second stage, this aligned red-green image acts as the reference or target to align the blue camera image using the same process as before. Afterwards, the aligned blue image is ready to be combined with the other two images to form a coarse aligned color (RGB) image as shown in Fig. 4. A key thing to note here is that this method of coarse alignment of the images is translationally and rotationally invariant. Specifically, this method does not depend on the distance or relative orientation between the cameras. As a result, this alignment process is robust to minor changes in the relative position or orientation of the CMOS sensors during the manufacturing process.

Fine Alignment:

Next, we perform fine alignment to correct artifacts and infer the missing color information. We designed a convolution network, SeaNet, consisting of 2D convolutional layers with skip connections. Fig. 5 shows the end-to-end neural network architecture. SeaNet consists of 10 2D convolution layers($\sim 81k$ parameters), 2 max-pooling layers and 2 upsampling layers. Upsampling layers help in changing the activation size before the skip connections. We have observed that these skip connections are critical in preserving the structural information of the image that is lost at the convolution and pooling steps. We give the course-aligned image (from the previous step) as input to the network. Although this image is not an accurate representation of the final color image, it has enough information for the network to reconstruct a representative color image.

To train our network, we use NeRF-Stereo Dataset [56] with an 80-20 split (on $\sim 6k$ image triplets). This dataset comprises trinocular real-world image triplets with 3 different camera distance ranges. We divide these image triplets into left, center, and right images. Since, in SeaScan, our idea is to use 3 passive filter cameras where the leftmost camera is the red camera, the middle camera is the green camera, and the rightmost camera is the blue camera; we extract the red, green, and blue

[3] In principle, one could use any of the other cameras as reference.

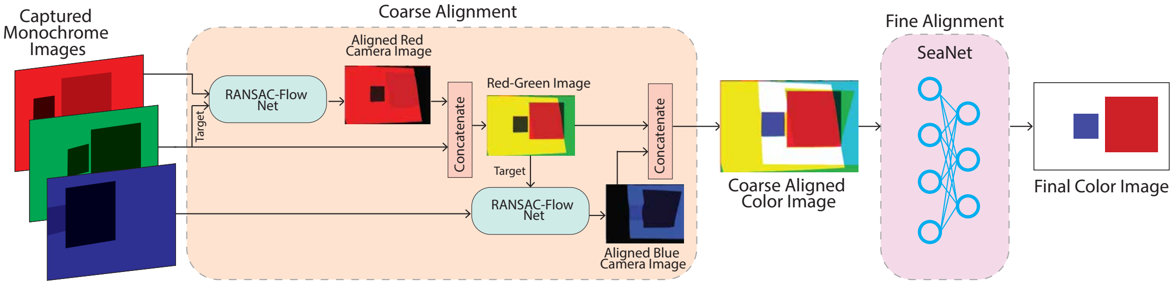


Figure 4: Cross Color Alignment: works in two steps 1) Coarse Alignment and 2) Fine Alignment. In Coarse Alignment, RANSAC-Flow Net coarsely aligns the images in a pairwise fashion. This coarse aligned color image then goes through the Fine Alignment step where SeaNet produces the final color image

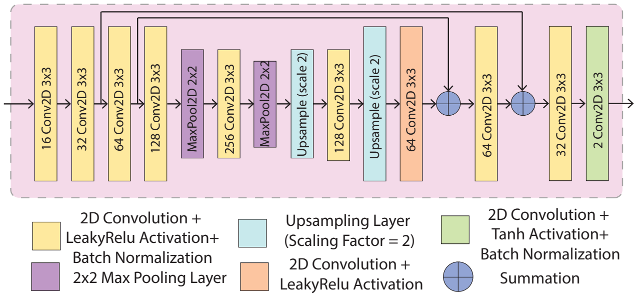


Figure 5: SeaNet Architecture: The Conv2D layers are written following the convention *[channel] Conv2D [k × k]* where *channel* is the number of output channels and *[kernel × kernel]* is the kernel size

channels from the left, center, and right images of the dataset, respectively. We then convert these images into grayscale images before feeding them to the neural network. The color images corresponding to the center camera are used as ground truth to train SeaNet. During the process of training, these images were randomly reshaped to improve the performance of the neural network on different sizes of images.

3 CROSS-REFRACTIVE COMPENSATION AND FILTER SELECTION

Up to this point, we have described how our *cross-color alignment* technique works and how we use a SeaNet to correct for missing colors in the coarse aligned images. In this section, we will discuss the complementary design of both the underwater encapsulation and compensation for the unique lensing and distortion effects experienced in underwater environments.

3.1 Distortion Compensation

While our CMOS imaging sensor performs excellently in terms of power, its wide-angle lens introduces a radially symmetric negative distortion in the resulting images. This distortion is known as barrel distortion [35]. To illustrate this, Fig. 6a shows a checkerboard pattern captured from the CMOS imaging sensor placed at roughly 30 cm from the sensor against a straight wall in air. The farther away we look from the center of the image, the more the distortion increases, and the straight lines of the checkerboard begin to curve into an increasingly barrel shape. As a result, the image magnification decreases with the distance away from the center.

For our cross-color alignment approach to work properly

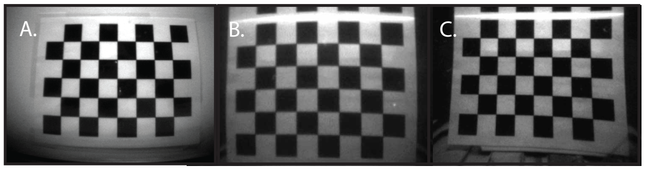


Figure 6: Distortion Effects: (a) A checkerboard in air with distortion. (b) A checkerboard underwater with a dome encapsulation. (c) A checkerboard underwater with flat encapsulation.

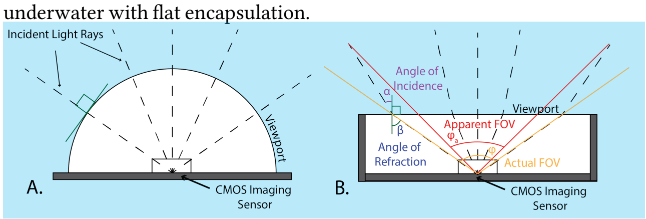


Figure 7: Encapsulation Methods: (a) Dome encapsulation that passes incident light rays without refraction. (b) Flat encapsulation that decreases the FOV and compensates for the wide-angle effect of the image sensor

(similar to any learning-based approach), the underlying distributions of training and testing data need to be similar. However, due to the barrel distortion, this relationship between our training data (described in §2.2) and our test data (from the real world) is mismatched, severely impacting the performance of our cross-color alignment algorithm on the raw images.

One possible solution to this problem is to augment the training dataset to include these distortions via modeling the imaging sensor. However, this approach requires careful characterization of the CMOS sensor [23] and the optical properties of its lens. Unfortunately, not only do we not have access to the exact lens shape and its optical characteristics, but mathematically modeling the interaction of the lens with the scene to sufficiently represent the true distribution of the testing dataset is challenging and prone to significant errors. Furthermore, simply removing the lens is not a viable option (it would result in an unfocused image) and custom-designing a distortion-compensation lens is less desirable than relying on commercial off-the-shelf components.

Thus, to correct for distortions without introducing significant complexities, our idea is to leverage underwater light propagation properties. Specifically, we know that when light rays travel from one medium to another they experience refraction. The direction of this refraction depends on both the refractive indices of the two mediums and the angle of incidence. We can exploit this refraction between the two mediums to cancel out the barrel distortion in the captured images. In what follows, we describe two different encapsulation designs and their impact on the observed barrel distortion.

Dome Encapsulation: One of the more common encapsulations for underwater imaging systems is a dome as shown in Fig. 7a. In such encapsulations, the rays of light entering are perpendicular to the tangential plane and parallel to the normal vector at that point, allowing the light to focus on the center of the dome without any refraction. As a result, this type of encapsulation does not introduce any change in the distortion pattern of the CMOS imaging sensor if the camera is placed at the center of the dome. Fig. 6b shows an image taken underwater from the CMOS imaging sensor when placed in a dome encapsulation. Since the dome doesn't perform any distortion compensation, the barrel distortion in the checkerboard still exists and does not cancel the effect of the wide angle lens.[4]

Flat Encapsulation: We propose to use a flat encapsulation to correct the distortion in the images captured from the CMOS imaging sensors. In this case, the light is no longer parallel to the surface normal. Fig. 7b shows how a flat acrylic housing causes the incident light to refract across the encapsulation boundary.[5] This can be mathematically written as:

$$\frac{n_{air}}{n_{water}} = \frac{sin(\alpha)}{sin(\beta)} \tag{1}$$

where n_{air} and n_{water} are the refractive indices of air and water; α and β are the angles of incidence and refraction. As a result, the field-of-view (FOV) of the imaging sensor is decreased, nearly eliminating the wide-angle distortion caused by the lens. Fig. 6c. shows the image of the same checkered board captured by the CMOS sensor when placed in a flat encapsulation.

To quantitatively evaluate the effectiveness of this method, we compute the distortion in the images before and after distortion compensation as $D = \frac{\Delta H}{H} \times 100$, where H is the predicted height of the object in the image and ΔH is the change in the object's height from the predicted height in the captured image. Using this metric, we quantify that the flat encapsulation reduces the distortion by 4× improvement (from 8% to 2%). This resulting distortion is sufficiently small that it enables the trinocular cross-color alignment to work correctly in the end-to-end reconstruction, as we demonstrate qualitatively and quantitatively in §6. This illustrates that the barrel effect is significantly reduced in the captured image and enables our test dataset to be representative of and drawn from the same distribution as the training dataset.[6]

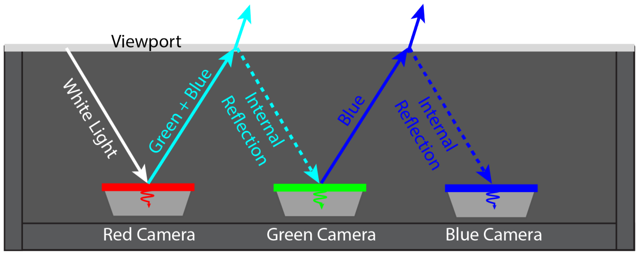


Figure 8: Dichroic Filter Problem: Incoming white light reflecting from dichroic filters, introducing artifacts into the captured images.

3.2 Filter Selection

Next, we focus on the selection of the filters for the red, green, and blue channels. Recall from §2 that our idea is to use a passive colorization technique using color filters on three grayscale cameras to enable energy-efficient color imaging. Proper performance of the color filters is critical to the end-to-end colorization performance of SeaScan.

There are two key factors to be considered in the filter selection. First, the wavelength bands of the filters should not have significant overlap. This is because we need the monochrome images captured by each camera to be as orthogonal in color (to the human eye) as possible to each other. If the wavelength bands have significant overlap, then colors that have significant power in multiple images can recombine to an incorrect output image color. Second, we must also make sure that the presence of the filters themselves does not introduce artifacts such as distortions, internal reflections, or blurriness in the captured images. These artifacts can both deteriorate the performance of our cross color alignment method and effect recovered colors in the images. Given these considerations, we investigated two types of filters: dichroic filters and gel filters.

Dichroic Filters: Dichroic filters are highly precise color filters that only pass light of very specific wavelengths [61]. They operate on the thin-film interference phenomenon, and are widely used in optical research, high-power laser applications, and even theatrical lighting [66]. While these filters perform very well at filtering specific wavelengths, they *reflect* wavelengths outside of their filtering range. This poses a problem for SeaScan because the unwanted wavelengths don't actually disappear. In fact, shown in Fig. 8, the wavelengths reflected from any filter partially reflect back off the viewport and can impinge onto a different filter that will pass the wavelength to the image sensor. This effect manifests by introducing significant lighting and reflection artifacts into

[4] The magnitude of barrel distortion seems less magnified in Fig. 6b. as compared to Fig. 6a. This is because when the CMOS sensor was placed in the encapsulation it was a little higher than the center of the dome because of the circuitry which changes the light propagation pattern. The higher we place the camera the more the encapsulation starts to behave like a flat surface

[5] The refractive index of clear acrylic is close to that of the water therefore for simplicity, we can assume the acrylic and water to be one medium and air as a different medium.

[6] Our discussion in this section has focused on one camera lens, however, this also extends to our three-camera imaging system.

each of the images. This creates errors in the depth estimation process and negatively affects the performance of our cross-color alignment method.

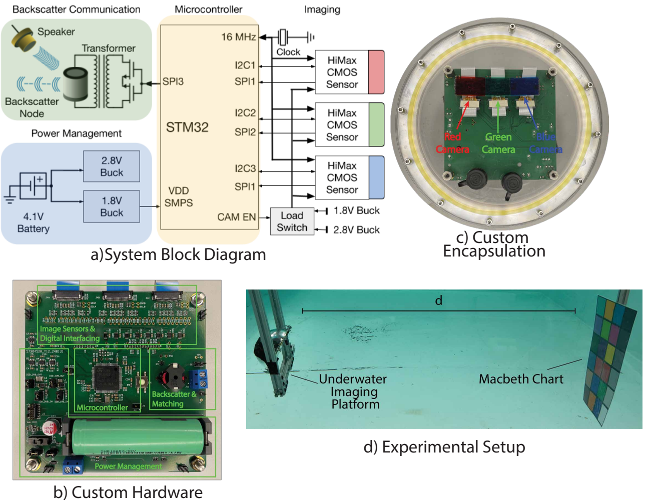
Gel Filters: A color gel or a lighting gel is a simple transparent colored material that is used to change the color of any white light [32]. The primary advantage of these filters is that they operate by *absorbing* unwanted wavelengths rather than totally reflecting them. In the context of our encapsulation, this means that the filters will absorb wavelengths that should not pass through the filter and suppress the lighting and reflection artifacts that are seen with dichroic filters. Even though these filters don’t have well-defined passband wavelengths like the dichroic filters do, their band of operation and filtering performance are sufficient to reconstruct color images from the monochrome image sensors. Thus, we selected gel filters as our red, green, and blue color filters in SeaScan’s design.

4 END-TO-END SYSTEM

Now that we have described SeaScan’s core design, we describe how we extend it to an end-to-end system.

From 2D to 3D: 3D underwater color reconstruction is important for various applications: monitoring fish growth, infrastructure, etc. Thus, we investigated various methods that can leverage monocular and stereo cameras for 3D reconstruction [11, 21, 49]. Our investigation showed that a (very) recent depth estimation method, called Depth Anything [65], performs extremely well. This method takes monocular images as input and outputs the depth estimation for each pixel in the captured scene. Thus, we integrated Depth Anything into our end-to-end pipeline. Specifically, SeaScan feeds the 2D color images to Depth Anything to estimate the depth of each pixel, then combines the depth estimates with the captured color image to reconstruct a 3D point cloud of the environment.

Compression: Recall that SeaScan aims to transfer its captured images wirelessly to a remote receiver. Unfortunately, the limited bandwidth of underwater acoustic communication (kHz-level) leads to long data transfer durations (e.g., around over an hour for the state-of-the-art low-power underwater camera), and can consume up to 20% of the overall energy of the system (even if it relies on an ultra-low-power underwater communication technology like backscatter). Thus, to further improve the energy efficiency of the system, we implement an on-board JPEG compression algorithm to reduce the size of the image that needs to be communicated. The on-board JPEG compression algorithm helps us reduce the image size significantly. By controlling the quality of the compressed image (a tunable parameter in the JPEG compression algorithm), we can control the amount of data that needs to be transmitted. This allows us to considerably reduce the communication time as compared to the state-of-the-art system which improves the overall energy efficiency.



a)System Block Diagram

c) Custom Encapsulation

b) Custom Hardware

d) Experimental Setup

Figure 9: System Implementation & Evaluation Overview

Backscatter Communication: After capturing and compressing the filtered monocular images, SeaScan needs to wirelessly transmit this data to a remote server for color recovery and 3D reconstruction. Since SeaScan’s goal is to enable energy-efficient underwater imaging, it integrates underwater backscatter [29] for net-zero power wireless communication. Specifically, the pixel data for each monochrome image is encoded into data frames and transmitted on an uplink channel. Similar to past work, the data frame can include headers (with sequence numbers and addressing), footers (CRC checks), and coding. At the physical layer, the data is encoded using FM0 modulation, and the backscatter is governed by an embedded microcontroller that interfaces with the backscatter switch. Our design focused on omnidirectional backscatter nodes, but can be extended to retroreflective designs that would enable data transmissions up to hundreds of meters as demonstrated in recent work [3, 4]. Furthermore, if desired, the design can be extended via energy harvesting [2, 29] to enable battery-free operation of SeaScan.

5 IMPLEMENTATION

5.1 Embedded Hardware

We designed and implemented a custom printed circuit board (PCB) powered by a Samsung 25R 2500 mAh battery [1] to house the three cameras, a microcontroller for receiving, processing, and transmitting the images, and backscatter switching. The overall schematic architecture is shown in Fig. 9a and the PCB is shown in Fig. 9b.

(a) Image Sensors We use the HM01B0 CMOS imaging sensors [23] from Himax as our 3 cameras. These cameras consume a maximum of 4 mW during the active capture phase.

(b) Microcontroller Our microcontroller is STM32U535VE-TQ6 from STMicroelectronics [38]. We use its serial peripheral interfaces (SPI) to receive the serial data from the HM01B0 cameras. We chose this microntroller for its superior ultra-low-power consumption in both run and low-power modes. We send a main clock (MCLK) frequency of 4 MHz to each of the cameras and sequentially take the pictures from camera 1, camera 2, and camera 3 by receiving the serial data on two of the three available SPI interfaces: SPI1 and SPI2. These SPI interfaces are placed in the receive-only slave mode and use the HM01B0 cameras' pixel clock (PCLK) output as their SPI clocks. We capture the images by receiving data from camera 1 on SPI1, from camera 2 on SPI2, and then we hot-swap the SPI1 to a different set of pins to receive data from camera 3. Note that the three cameras capture images sequentially due to the limited availability of the SPI on the STM32U535VETQ6 microcontroller; however, the three cameras can still capture images in a fraction of a second. Since this microcontroller contains 274 KBytes of SRAM, we can store each of the 320 × 240-sized images temporarily in SRAM before JPEG compression.

(c) JPEG Compression & Backscatter Transmission Once the images are captured, we JPEG compress each of them using the open-source JPEC library [39] using a quality factor of 30% to prepare them for backscatter transmission. The CPU operates at an 80 MHz clock rate in order to compute faster and optimize the energy consumption of the system. After compression, we throttle the microcontroller's clock rate down to the minimum possible 100 kHz for the lowest possible power consumption. We transmit the compressed image data to the backscatter switch using the SPI3 interface in transmit-receive slave mode. The image data bits are encoded into FM0 symbols which are then transferred to the SPI3 transmit data buffer via the direct-memory access (DMA) controller. Backscatter switching is realized using a dual n-channel BSD840NH6327XTSA1 [52] MOSFET.

5.2 Encapsulation

Recall from our discussion in §3, our camera needs to be encapsulated such that the CMOS imaging sensors face a flat clear acrylic housing. To do so, we laser cut a clear, scratch-resistant piece of acrylic into a circle that serves as the viewport for the scene. This circular viewport fits the circumference of a dome[7] that provides enough room to house our custom PCB and image sensors. The acrylic viewport and backing dome are secured together with 8 machine screws that compress an o-ring to create a watertight seal. We attached a Micro-Con-X [15] 4 pin circular connector to the flat viewport in the encapsulation as shown in Fig. 9c.

5.3 Software

The camera data was received and decoded in MATLAB on an Ubuntu 20.4 machine. This machine is connected to a remote server with a GTX 1080 Ti GPU where we perform Cross Color Alignment, depth estimation and point cloud generation. We trained SeaNet with the GTX 1080 Ti GPU in Python with Adam optimizer and smooth L1 loss for 300 epochs.

6 RESULTS

6.1 Experimental Setup

To evaluate the performance of SeaScan, all real-world experiments were performed underwater in either a small $20'' \times 12'' \times 10''$ tank or a large $4m \times 3m \times 1.2m$ experimental pool under standard indoor lighting conditions. We note that in all of our experiments, there was no underwater light present. We used a GoPro 9 camera to obtain the ground truth. In our evaluation, we use models of fish and coral reefs, as well as a Macbeth chart as a gold standard for color reconstruction accuracy [47]. Our evaluation setup with the Macbeth chart is shown in Fig. 9d.

6.2 Energy Consumption

We measure the energy consumption of SeaScan in three phases: image capture, data compression, and backscatter communication. For a given phase, as the microcontroller firmware runs, we measured the current trace of our system at different I_{dd} testpoints on the PCB using the DMM6500 multimeter [53]. We calculated the average power as defined by $P = I_{dd}V_{dd}$, where the V_{dd} is the voltage at the testpoint. Using the average power and the time duration, we computed the total energy consumption of the entire system and its components (as $E = \int P(t)dt$).

Tab. 1 compares the energy consumption to a state-of-the-art baseline for low-power underwater color imaging [2] for each of the three phases. We make the following remarks:

- The energy consumption during the image capture phase for SeaScan is 23.6 mJ, while that of the baseline is 894.2 mJ. This is a 37× reduction compared to the prior work.
- SeaScan takes 0.6 seconds for the image capture phase, and baseline requires 111.2 seconds. Thus, our system is able to capture images 185 × faster than the prior art. This is because our architecture runs at a much higher clock rate.
- For the JPEG compression phase, SeaScan consumes a total of 9 mJ.[8] However, due to the compression[9], the total energy for backscatter communication is significantly less for SeaScan (85.1mJ) than it is for the baseline (234.9mJ). Note that for fair comparison, we use the same datarate as in [2] i.e., 1kbps for the backscatter communication.
- SeaScan consumes a total energy of 117.7 mJ, while the baseline consumes 1129.1 mJ. Thus, our system achieves more

[7] This dome is NOT the viewport, it simply is a backing piece that creates the volume in which our electronics reside.

[8] The baseline does not perform image compression.

[9] We use a quality factor of 30% for JPEG compression. Although, the overall compression ratio depends on the information content of the raw image. In practice, we have observed that the output image can be anywhere from 3 - 6 KB in size with a 30% quality factor, yielding a compression ratio of between 12.8:1 and 25.6:1

	Image Capture Phase			JPEG Compression Phase			Backscatter Communication Phase			
	Average Power	Time	Energy	Average Power	Time	Energy	Average Power	Time	Energy	Total Energy
Baseline [2]	8.0 mW	111.2 s	894.2 mJ	—	—	—	59.0 μW	3975.0 s	234.9 mJ	1129.1 mJ
SeaScan	39.3 mW	0.60 s	23.6 mJ	39.3 mW	0.23 s	9.0 mJ	457.3 μW	186.1 s	85.1 mJ	117.7 mJ

Table 1: Energy Consumption. The table shows the energy consumption of the state-of-the-art imaging system [2] and SeaScan in Image Capture Phase, JPEG Compression Phase, and Backscatter Communication Phase.

	Image Capture Phase		JPEG Compression Phase		Backscatter Communication Phase	
Components	**Average Power**	**Energy**	**Average Power**	**Energy**	**Average Power**	**Energy**
Monochrome CMOS Sensors	3.8 mW	2.3 mJ	0 mW	0 mJ	0 μW	0 mJ
Microcontroller	28.9 mW	17.3 mJ	33.4 mW	7.7 mJ	395.3 μW	73.6 mJ
Other Active Components	6.7 mW	4.0 mJ	5.7 mW	1.3 mJ	61.8 μW	11.5 mJ
Total	39.3 mW	23.6 mJ	39.3 mW	9.0 mJ	457.3 μW	85.1 mJ

Table 2: Energy Consumption Breakdown. The table shows the energy consumption of different components of SeaScan over all three phases.

than 9× reduction in total energy consumption compared to the state-of-the-art underwater imaging system.

- Interestingly, SeaScan consumes more instantaneous power than the baseline. During the image capture phase, it has an average power consumption of 39.3 mW compared to 8 mW for the baseline. For the backscatter communication phase, SeaScan has an average power consumption of 457.3 μW, while the baseline has a power consumption of 59.0 μW. The higher power consumption of our system is despite the fact that SeaScan does not flash LEDs and is because SeaScan runs at a higher clock rate and has a larger memory block as compared to [2]. This was a deliberate design decision because a high clock rate is more energy-efficient even if it consumes more instantaneous power.

Tab. 2 breaks down the energy consumption of SeaScan by hardware components across the three phases. Here we note:

- The monochrome CMOS sensors require an average power of 3.8 mW during the image capture phase, and consume 2.3 mJ of energy. Note that the cameras are disabled during the JPEG compression and backscatter communication phases, hence they do not consume energy in those phases.
- The STM32U535VETQ6 microcontroller uses an average power of 28.9 mW during image capture, 33.4 mW during JPEG compression, and 395.3 μW during backscatter communication. The microcontroller consumes 17.3 mJ of energy over image capture, 7.7 mJ over compression, and 73.5 mJ over backscatter. The reduction in power consumption from JPEG compression to backscatter is caused by the microcontroller switching to a low power mode for the backscatter phase.
- The PCB also includes power converters and load switches which draw power during operation. These miscellaneous active components use 6.7 mW, 5.7 mW, and 62 uW during image capture, JPEG compression, and backscatter phases, respectively. The components consume 4 mJ, 1.3 mJ, and 11.5 mJ while collecting images, compressing data, and communicating over backscatter, respectively.

Note that SeaScan can capture color images without the need to illuminate the scene with different color lights. Furthermore, it can be augmented with a white LED (10.73mJ) to enable its operation in dark environments (e.g., in deep sea). In this case, the energy consumption of SeaScan during the image capture phase increases to 23.6 mJ + 3×10.73 mJ = 55.79 mJ, which is 16× lower than the 894.2mJ consumed in [2]. Here, the 16× improvement comes entirely from SeaScan's novel optimized microcontroller architecture for ultra-low-power underwater imaging.

6.3 Qualitative Results

Next, we evaluate the qualitative performance of SeaScan and compare it to alternate implementations and a GoPro camera underwater as ground truth.

Alternate Implementations. We compared the performance of our *Cross Color Alignment* algorithm with (recall from §2)

- *Direct Combination:* This method involves directly applying red, green, and blue images to the red green and blue channels of a color image without any transformation.
- *Linear Transformation Alignment:* Here, we use the position of the three cameras to determine the transformation between them. Then, we linearly invert these transforms before applying the three images to their respective channels.
- -based Colorization-based Colorization-based Colorization-based Colorization*NN-based Colorization:* [68] This baseline uses state-of-the-art neural network based colorization method to convert grayscale images into color images.

Imaging Results. Fig. 10 shows real images captured by our setup from multiple scenes.

The first column in Fig. 10 shows the image captured by the GoPro 9 camera. The second, third, fourth, and fifth column of Fig. 10 show the color images as a result of Direct Combination,

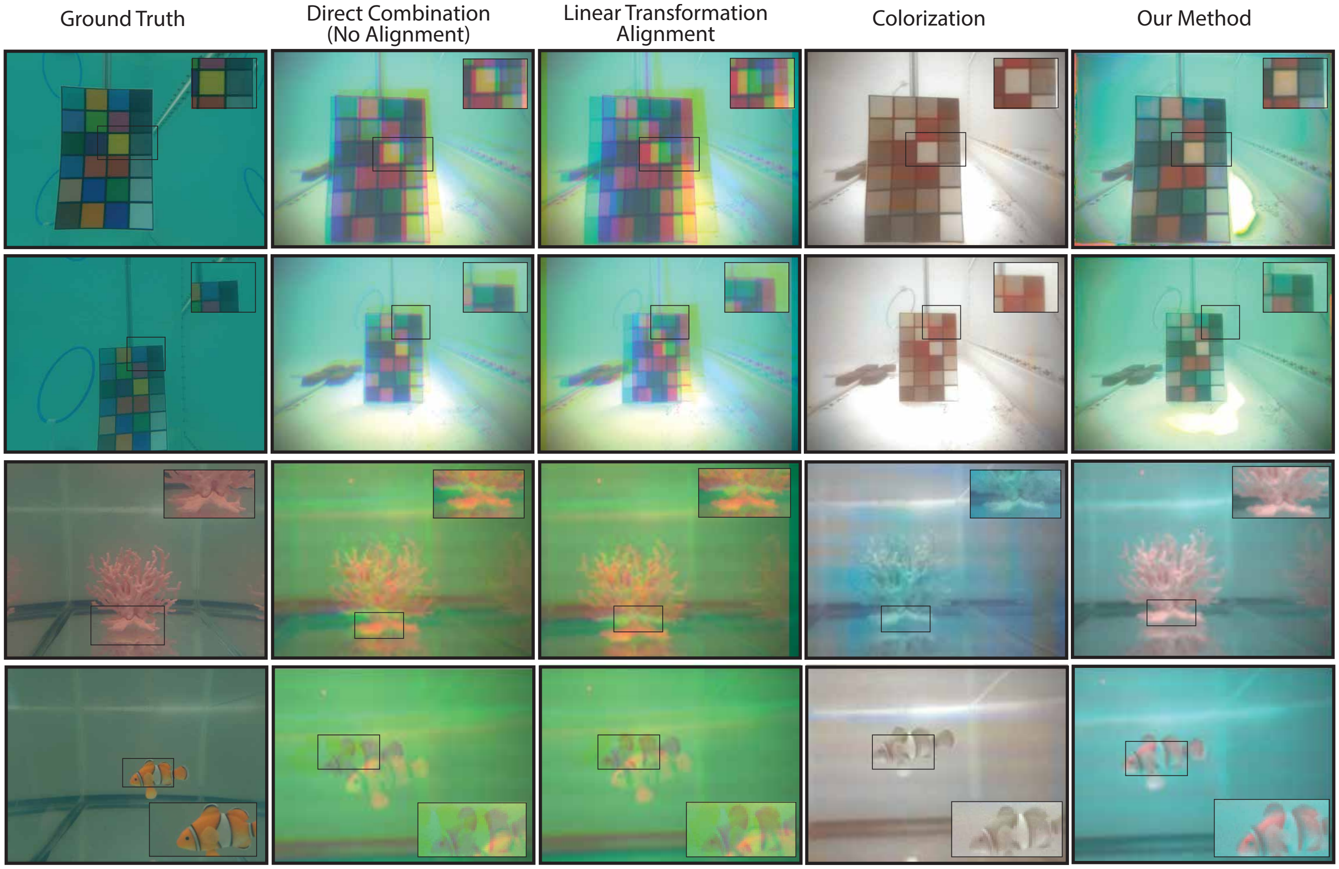
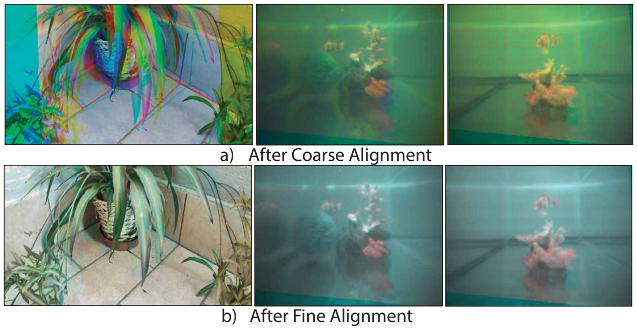

Ground Truth
Direct Combination (No Alignment)
Linear Transformation Alignment
Colorization
Our Method

Figure 10: Qualitative Results. This figure shows the qualitative results of our method and the baselines. The first column shows the images taken from the GoPro 9 camera. The columns represent different imaging methods and the rows represent different scenarios The second, third, and fourth columns show the results of Direct Combination, Linear Transformation Alignment, and NN-based Colorization, respectively. The fifth row shows the result of our system.

Linear Transformation Alignment, NN-based Colorization, and our method respectively. The first row shows the set of images for a Macbeth chart taken from a distance of $70cm$ from the camera. The second row shows the set of images of the same object in the same environment with a distance of $100cm$ from the camera. The third and fourth rows show results for an image of a plastic model of a coral reef and a toy fish, respectively, captured from a distance of 30 cm. Based on these qualitative results, we note:

- It can be seen from Fig. 10 that Cross Color Alignment (our) method perfectly aligns the three images taken from three different cameras and fills in the information gap as well while keeping the true colors of the scene intact.
- Note that Direct Combination performs better in the image of the Macbeth chart at 1m (second row) compared to 0.7m(first row). This is because as we move the object further from the camera, the difference in the position of the object in the three camera images is very small, reducing the need for alignment for far away objects. However, when zoomed into the edges of the chart in the second row, one can notice the blurry edges and artifacts which are the result of misalignment between the three cameras.
- Linear Transformation Alignment is not able to align the colors in different scenarios, this is because this method uses the physical transformation between cameras. As explained in §2, this physical transformation does not match the transformations between the image of an object across the three camera frustum.
- Note that NN-based Colorization fails in coloring the images, especially the Macbeth charts. This is because the neural network, which NN-based Colorization relies on, guesses colors based on visual cues in images, and there is no reliable cue to enable guessing Macbeth chart colors.

Impact of Alignment Steps. Fig. 11 shows the qualitative performance of Cross Color Alignment algorithm steps. Recall from §2.2, Cross Color Alignment is a two-step process. Fig. 11a shows the qualitative performance of Cross Color Alignment after the first step (i.e., Course Alignment), and Fig. 11b shows the results after the second step (i.e., Fine Alignment). The first column from the left in Fig. 11a and b is from the test dataset [56] and the other two images are captured from our energy-efficient imaging platform. After the coarse


a) After Coarse Alignment

b) After Fine Alignment

Figure 11: Impact of Cross Color Alignment steps. a) shows the impact of Coarse Alignment b) shows the performance of the Fine Alignment step.

alignment step, most of the objects in the Fig. 11a images are misaligned with artifacts at the edges and incorrect colors in some parts of the images. Once we apply fine alignment on these images, note that the fine alignment step (i.e., the SeaNet) has removed the misalignment and the colors of the images are more representative of the captured scenes.

6.4 Color Classification

In this section, we evaluate the performance of SeaScan in detecting different colors in each scene. This is important since many underwater imaging applications rely on tracking the changes in the appearance of underwater objects over time, for instance, to detect coral reefs' bleaching by observing their color change over time [18], or the changing hues in colors of marine mammals as a result of diseases and infections [20].

To perform color classification, we first transformed the colors of the pixels from RGB space to LAB color space, a three-dimensional color space that is focused on the human color perception range [9]. We then perform KNN classification on $n \in \{3,4,5,6,7,8,9\}$ different colors (furthest colors in LAB space that are present in the macbeth chart).

Fig. 12a plots the accuracy of the color classification algorithm on the output of our method and the baselines. The figure also plots the expected accuracy of a random guess where it randomly assigns a color class to each pixel. We note:

- SeaScan consistently outperforms all the baseline and shows a 100% accuracy in classifying colors into 3 categories
- Note that although Direct Combination reached 91% accuracy in classifying pixels into 3 classes, its accuracy drops below 35% when the number of classes is increased to 9.
- The Linear Transformation Alignment shows 80% accuracy when there are only 3 classes, and drops to 22% accuracy with 9 classes. The random policy has 11% accuracy with 9 classes.
- NN-based Colorization has 55% accuracy with only 3 classes, and its accuracy falls to 20% with only 8 classes.

6.5 PSNR vs Distance

To evaluate the performance of our system at different distances, we capture images of the Macbeth chart at different distances from our system and calculate the Peak Signal-to-Noise Ratio (PSNR) using the images captured from a GoPro 9 camera as ground truth. PSNR is a standard quantitative measure to evaluate the quality of an image in the presence of noise. Higher PSNR indicates a better quality image. Note that the physical position, camera resolution, and camera instrinsics of GoPro and SeaScan are different. Consequently, pixel-by-pixel comparison of the images captured from GoPro and SeaScan is not possible. To deal with this problem, we extracted the individual Macbeth chart color patches from the GoPro and SeaScan images manually and compare them. Therefore, it is the relative PSNR values that are important here rather than the absolute values.

Fig. 12b plots the image's PSNR on the y-axis and the distance of the Macbeth chart from the camera on the x-axis. We make the following remarks:

- Note that the PSNR of our method (purple line) is on average $2dB$ higher than the simple alignment method at all distances. This is considered a meaningful quantitative improvement in the imaging literature [12, 37, 54], yet the qualitative results reported earlier remain the more desirable approach to evaluate imaging performance.
- The performance of simple alignment and linear transformation alignment methods show very similar trends across different distances with an average difference of $\sim 0.5dB$

6.6 PSNR vs Image Quality (JPEG)

In this section, we evaluate the performance of SeaScan against different image qualities as a result of image compression. Recall from §4 that we implement image compression on our system to reduce the data that needs to be transmitted to further decrease the energy consumption of our system.

Fig. 12c shows the PSNR on the y-axis and JPEG image quality on the x-axis for our method and the baselines. Note that even at 10% image quality the performance of our method has negligible effect and consistently performs better than the baselines. Our method consistently has a PSNR above 9.5 dB, while baselines consistently have a PSNR lower than 8.5 dB.

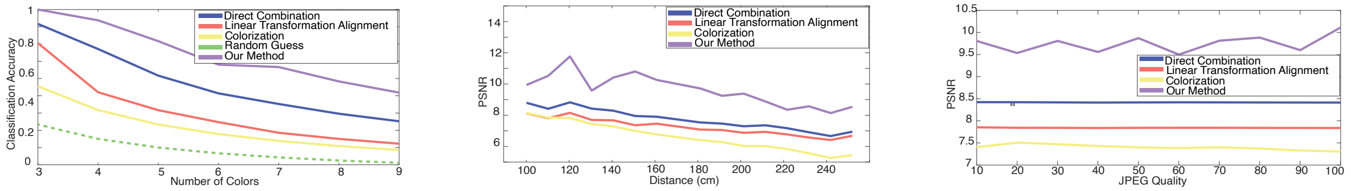
Fig. 13 shows the qualitative results of our method on three image qualities(30%, 60% and 100%). Note that the performance of Cross Color Alignment method is not affected even though the quality of the image has been significantly reduced.

6.7 Performance of Passive Colorization

To further evaluate the performance of the Cross-Color Alignment algorithm, we use two types of datasets 1) unseen online test dataset [56] and 2) real-world images taken with SeaScan.

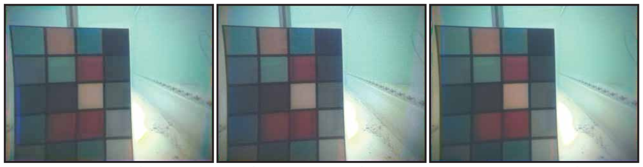
In addition to PSNR, we compute two additional metrics:

- *Structural Similarity Index [8]*: SSIM is an image quality metric between -1 and 1 indicating the structural similarity between any given two images. Similar to PSNR, the higher the SSIM better the performance of the system.



(a) Classification Accuracy **(b) PSNR vs Distance** **(c) PSNR vs Image Quality**

Figure 12: (a) shows the color classification accuracy vs the number of colors (or classes). (b) shows the performance of our method and the baseline at different distances, plotting PSNR vs distance. (c) shows the effect of the compression on the performance of our method and the baselines, ploting PSNR vs image quality.



a)30% Image Quality b) 60% Image Quality c) 100% Image Quality

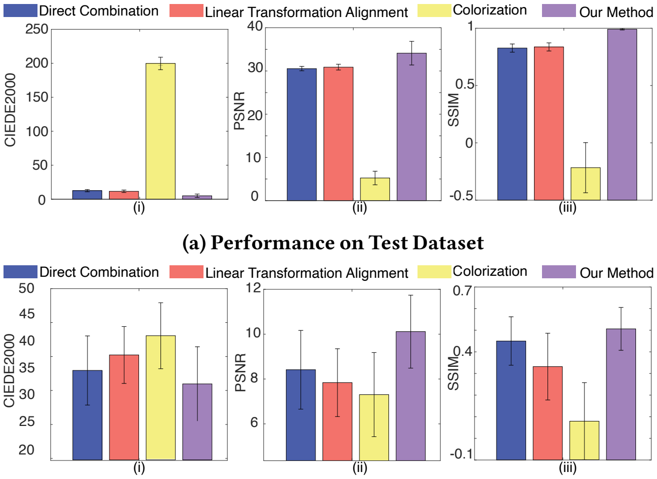
Figure 13: Performance on Different Image Qualities. SeaNet's performance on images with 30%, 60%, and 100% JPEG compression qualities

- *CIEDE2000 [22]*: A quantitative metric to measure the perceived color difference between two images. Unlike PSNR and SSIM, lower CIEDE2000 is better.

Performance on Online Test Dataset. We evaluate the performance of our Cross-Color Alignment algorithm, described in §2, on unseen online dataset [56]. Specifically, this dataset includes multiple RGB images from the same scene. To evaluate our system on this dataset, we choose 3 different images from each scene in the test dataset, and extract the red channel from the first, green channel from the second, and the blue channel from the third image. This allows us to simulate SeaScan images from its three red, green, and blue camera.

We evaluate the performance of our algorithm on over 1190 images. Fig. 14a shows the performance of our alignment method and compares it to the baselines as bar plots. Specifically, Fig. 14a shows the average CIEDE2000, PSNR, and SSIM respectively for our method (purple bars), NN-based Colorization baseline (yellow bars), Direct Combination baseline (red bars), and Linear Transformation Alignment baseline (blue bars). We make the following remarks.

- CIEDE2000 (Fig. 14a(i)) of our method is 5.05, ~ 2× less than the simple and linear transformation alignment methods.
- CIEDE2000 for NN-based Colorization baseline is much higher than that of the other three methods with a value of 199.7. This is because, out of all four methods we are comparing, NN-based Colorization method does not have any hint about the color of the objects in the scene. It relies solely on the neural network parameters learned during training.
- PSNR (Fig. 14a(ii)) of Direct Combination and Linear Transformation Alignment are comparable with values of 30.5 and 30.8dB respectively. Whereas our method shows better performance with a PSNR of 34.1dB.
- Fig. 14a(iii) shows the SSIM of the four methods. It is to be noted that SSIM ranges from −1 to 1, with its value being 1 for two identical images. Note that our method shows an SSIM of 0.9913 which is very close to a perfect SSIM, wheres the other three methods show SSIM of 0.82,0.83 and −0.22



(a) Performance on Test Dataset

(b) Performance on the Captured Images

Figure 14: Quantitative Performance of Passive Colorization. (a) and (b) plots CIEDE2000, PSNR, and SSIM performance of the SeaNet on test data [56] and underwater images captured by SeaScan, respectively.

Performance on Images Captured by SeaScan˙ Now we evaluate the performance of the Cross-Color Alignment algorithm on the images captured using our platform. We image the Macbeth chart, as shown in Fig. 9d, at different distances using our platform. We used a GoPro 9 camera and placed it next to our camera to capture the image of the same scene. As mentioned in §6.5, we cannot directly compare the two images captured by these two cameras. Therefore, we extract the color patches out of the images to perform the comparison. Fig. 14b shows the CIEDE2000, PSNR, and SSIM of our method and the baselines. We make the following remarks:

- Fig. 14b(i) shows that our method has a CIEDE2000 of 35.8, $\sim 5 - 20\%$ less than the baselines. It is to be noted that since these values are calculated by comparing the images captured using a low-power CMOS sensor with a high-resolution GoPro 9 camera, it is the relative difference that has more significance than the absolute value. This plot signifies the correctness of the color and is highly dependent on the choice of filters. With better filter design the absolute CIEDE2000 can be further reduced.

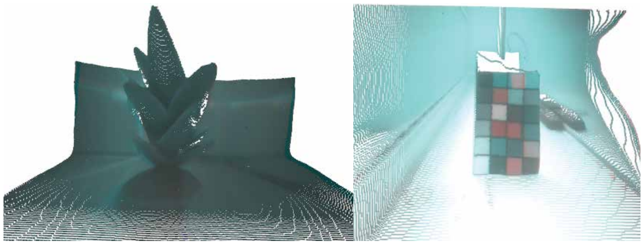


Figure 15: Estimated 3D Point Clouds. shows the estimated point cloud of a plant in a small tank and Macbeth chart in a large tank

- Fig. 14b(ii) shows the PSNR of our method in purple as compared to the other baselines. Our method shows an increase $\sim 2-3dB$ increase in PSNR as compared to the baselines.
- Fig. 14b(iii) shows the SSIM of our method and compares it with the SSIM of the baselines. It can be seen from the plot that the average SSIM of our method is over 0.5.

6.8 Generated Point Clouds

Finally, we qualitatively show sample 3D point clouds generated using the images captured from SeaScan in Fig. 15 for an underwater plant and Macbeth chart. This shows that SeaScan is capable of being used with the existing computer vision techniques to extend its applications.

7 RELATED WORK

Low-power & Underwater Cameras. The past few years have witnessed growing interest in ultra-low-power cameras in the mobile computing community [28, 30, 41, 43, 58]. Past work has achieved important advances in developing low-power imaging systems, adding wireless capabilities, and increasing their frame-rate and resolution. However, the majority of this past literature has focused on *grayscale* imaging *in air*. Our research shares the motivation of these past systems and aims to achieve ultra-low-power *color* imaging in *underwater* environments.

One might wonder whether recent systems for low-power color imaging in air – such as Video Backscatter [41] and NeuriCam [58] – could be used for underwater color imaging. However, Video Backscatter's COTS implementation is limited to grayscale imaging (the potential for color imaging), and it still requires specialized ASICs to achieve color imaging. Additionally, NeuriCam's approach would not work well underwater. This system combines a (highly duty-cycled) color sensor with a (high-frame rate) grayscale sensor; by streaming sensor high-frame-rate data to a receiver, it can interpolate the color images. However, due to the significant bandwidth limitations of the underwater acoustic channel, it remains infeasible today to stream high frame-rate videos using low-power underwater cameras. Moreover, this past system consumes 83 mW, which is significantly higher than SeaScan's peak power consumption.

In the context of underwater imaging, most past systems consume tens of Watts, which makes it difficult to use them for long-term battery-powered operation [7, 17, 34, 59, 60, 64]. The only system we are aware of for low-power underwater color imaging consumes around 1 J of energy due to LED flashing [2]. SeaScan is motivated by this recent system and has orders of magnitude lower energy consumption.

Finally, past work has investigated building custom chips for low-power imaging [10, 14, 33, 42]. However, since these ASICs are not commercially available, they require specialized costly fabrication. In contrast, SeaScan is designed entirely using COTS hardware, making it easily available for researchers and practitioners for use and deployment.

Colorization and Color Reconstruction. The problem of colorizing grayscale images has received attention over the past few years [5, 13, 62, 69], for example, to colorize footage of past historical events. The vast majority of these methods rely on ML-trained models for colorization. However, because these methods are trained on in-air images (mainly of humans and aerial sites), they do not perform well on underwater images as we demonstrated empirically in our results. Moreover, due to the dearth of available underwater footage, it is difficult to retrain for underwater environments.

There has also been past research on ML-based approaches for color enhancement of underwater color images to perform dehazing or extract the true colors of the scene [6, 19, 36, 57]. These methods address an orthogonal problem to SeaScan, and SeaScan's output can be fed into them to extract true color (similar to how an underwater GoPro image can be fed into them to extract its true colors).

8 CONCLUSION

Low-power underwater imaging is important for long-term observations of subsea environments, with applications ranging from ocean climate change monitoring and scientific discovery to seafood production and robotic navigation. This paper marks a new step towards that vision by introducing a highly energy-efficient underwater color imaging system. As the research evolves, it would be valuable to investigate methods that continue to push the boundaries of these systems through even lower power as well as higher frame rates and resolutions. Moreover, it would be interesting to investigate how some of the proposed methods here (e.g., trinocular lensing) may lend themselves to in-air imaging applications. More generally, the paper sits at the intersection of two emerging trends in the mobile computing community - of low-power imaging and ocean IoT - and we believe that the area of low-energy underwater imaging will benefit from technical advances in both of these areas.

Acknowledgments. We thank our shepherd, the anonymous reviewers, and the Signal Kinetics group for their feedback. We also thank MIT SeaGrant. The research is funded by the Office of Naval Research and National Science Foundation.

REFERENCES

[1] [n. d.]. Samsung 25R 18650 2500mAh 20A Battery. https://www.imrbatteries.com/samsung-25r-18650/.

[2] Sayed Saad Afzal, Waleed Akbar, Osvy Rodriguez, Mario Doumet, Unsoo Ha, Reza Ghaffarivardavagh, and Fadel Adib. 2022. Battery-free wireless imaging of underwater environments. *Nature communications* 13, 1 (2022), 1–9.

[3] Waleed Akbar, Ahmed Allam, and Fadel Adib. 2023. The Underwater Backscatter Channel: Theory, Link Budget, and Experimental Validation. In *Proceedings of the 29th Annual International Conference on Mobile Computing and Networking*. 1–15.

[4] Lester Clare Van Atta. 1958. Underwater sound reflector. Patent No. US-2839735-A, Filed June 8th, 1955, Issued Jun. 17th, 1958.

[5] Andrea Bejarano-Carbo, Hyochan An, Kyojin Choo, Shiyu Liu, Qirui Zhang, Dennis Sylvester, David Blaauw, and Hun-Seok Kim. 2022. Millimeter-scale ultra-low-power imaging system for intelligent edge monitoring. *tinyML Research Symposium* (2022).

[6] Dana Berman, Deborah Levy, Shai Avidan, and Tali Treibitz. 2020. Underwater single image color restoration using haze-lines and a new quantitative dataset. *IEEE transactions on pattern analysis and machine intelligence* 43, 8 (2020), 2822–2837.

[7] Stephanie M Bilodeau, Austin WH Schwartz, Binfeng Xu, V Paúl Pauca, and Miles R Silman. 2022. A low-cost, long-term underwater camera trap network coupled with deep residual learning image analysis. *Plos one* 17, 2 (2022), e0263377.

[8] Dominique Brunet, Edward R Vrscay, and Zhou Wang. 2011. On the mathematical properties of the structural similarity index. *IEEE Transactions on Image Processing* 21, 4 (2011), 1488–1499.

[9] Laurent Busin, Nicolas Vandenbroucke, and Ludovic Macaire. 2008. Color spaces and image segmentation. *Advances in imaging and electron physics* 151, 1 (2008), 1.

[10] Ismail Cevik and Suat U Ay. 2015. An ultra-low power energy harvesting and imaging (EHI) type CMOS APS imager with self-power capability. *IEEE Transactions on Circuits and Systems I: Regular Papers* 62, 9 (2015), 2177–2186.

[11] Jia-Ren Chang and Yong-Sheng Chen. 2018. Pyramid stereo matching network. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 5410–5418.

[12] Wei-Ting Chen, Hao-Yu Fang, Cheng-Lin Hsieh, Cheng-Che Tsai, I Chen, Jian-Jiun Ding, Sy-Yen Kuo, et al. 2021. All snow removed: Single image desnowing algorithm using hierarchical dual-tree complex wavelet representation and contradict channel loss. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 4196–4205.

[13] Zezhou Cheng, Qingxiong Yang, and Bin Sheng. 2015. Deep colorization. In *Proceedings of the IEEE international conference on computer vision*. 415–423.

[14] Kyojin D Choo, Li Xu, Yejoong Kim, Ji-Hwan Seol, Xiao Wu, Dennis Sylvester, and David Blaauw. 2019. 5.2 energy-efficient low-noise CMOS image sensor with capacitor array-assisted charge-injection SAR ADC for motion-triggered low-power IoT applications. In *2019 IEEE International Solid-State Circuits Conference-(ISSCC)*. IEEE, 96–98.

[15] Switchcraft Conxall. [n. d.]. Micro-con-X 2-pin Connector. https://www.innovasea.com/aquaculture-intelligence/.

[16] John B Corliss, Jack Dymond, Louis I Gordon, John M Edmond, Richard P von Herzen, Robert D Ballard, Kenneth Green, David Williams, Arnold Bainbridge, Kathy Crane, et al. 1979. Submarine thermal springs on the Galapagos Rift. *Science* 203, 4385 (1979), 1073–1083.

[17] Corrado Costa, Michele Scardi, Valerio Vitalini, and Stefano Cataudella. 2009. A dual camera system for counting and sizing Northern Bluefin Tuna (Thunnus thynnus; Linnaeus, 1758) stock, during transfer to aquaculture cages, with a semi automatic Artificial Neural Network tool. *Aquaculture* 291, 3-4 (2009), 161–167.

[18] AE Douglas. 2003. Coral bleaching—-how and why? *Marine pollution bulletin* 46, 4 (2003), 385–392.

[19] Akshay Dudhane, Praful Hambarde, Prashant Patil, and Subrahmanyam Murala. 2020. Deep underwater image restoration and beyond. *IEEE Signal Processing Letters* 27 (2020), 675–679.

[20] Salvatore Frasca Jr, Rebecca J Gast, Andrea L Bogomolni, and Steven M Szczepanek. 2020. Diagnosing marine diseases. *Marine Disease Ecology* 213 (2020).

[21] Clément Godard, Oisin Mac Aodha, Michael Firman, and Gabriel J Brostow. 2019. Digging into self-supervised monocular depth estimation. In *Proceedings of the IEEE/CVF international conference on computer vision*. 3828–3838.

[22] Cristina Gómez-Polo, María Portillo Muñoz, Mari Cruz Lorenzo Luengo, Purificación Vicente, Purificanción Galindo, and Ana María Martín Casado. 2016. Comparison of the CIELab and CIEDE2000 color difference formulas. *The Journal of prosthetic dentistry* 115, 1 (2016), 65–70.

[23] Himax. [n. d.]. HM01B0 Ultralow Power CIS. https://www.himax.com.tw/products/cmos-image-sensor/always-on-vision-sensors/hm01b0/. Accessed: 2024-03-12.

[24] Himax. [n. d.]. HM11B1 HD Sensor for Always On Vision Ultra Slim Camera Applications. https://www.himax.com.tw/products/cmos-image-sensor/always-on-vision-sensors/hm11b1/. Accessed: 2024-08-26.

[25] Michael Ho, Sami El-Borgi, Devendra Patil, and Gangbing Song. 2020. Inspection and monitoring systems subsea pipelines: A review paper. *Structural Health Monitoring* 19, 2 (2020), 606–645. https://doi.org/10.1177/1475921719837718

[26] Avnet Inc. 2023. *onsemi ARX3A0 CMOS image sensor ultra fast rolling shutter*. https://www.avnet.com/wps/portal/ebv/products/new-products/npi/2019/onsemi-ARX3A0/

[27] INNOVASEA. 2020. Aquaculture Intelligence. https://www.innovasea.com/aquaculture-intelligence/.

[28] Vikram Iyer, Ali Najafi, Johannes James, Sawyer Fuller, and Shyamnath Gollakota. 2020. Wireless steerable vision for live insects and insect-scale robots. *Science robotics* 5, 44 (2020), eabb0839.

[29] Junsu Jang and Fadel Adib. 2019. Underwater backscatter networking. In *Proceedings of the ACM Special Interest Group on Data Communication*. 187–199.

[30] Colleen Josephson, Lei Yang, Pengyu Zhang, and Sachin Katti. 2019. Wireless computer vision using commodity radios. In *Proceedings of the 18th International Conference on Information Processing in Sensor Networks*. 229–240.

[31] Kakani Katija, Rob E Sherlock, Alana D Sherman, and Bruce H Robison. 2017. New technology reveals the role of giant larvaceans in oceanic carbon cycling. *Science Advances* 3, 5 (2017), e1602374.

[32] Jennifer DT Kruschwitz. 1998. Optical Filter to Enhance the Lifespan of Colored Gels. In *Optical Interference Coatings*. Optica Publishing Group, ThE–4.

[33] Oichi Kumagai, Atsumi Niwa, Katsuhiko Hanzawa, Hidetaka Kato, Shinichiro Futami, Toshio Ohyama, Tsutomu Imoto, Masahiko Nakamizo, Hirotaka Murakami, Tatsuki Nishino, et al. 2018. A 1/4-inch 3.9 Mpixel low-power event-driven back-illuminated stacked CMOS image sensor. In *2018 IEEE International Solid-State Circuits Conference-(ISSCC)*. IEEE, 86–88.

[34] Katherine Lam, Robin S Bradbeer, Paul KS Shin, Kenneth KK Ku, and Paul Hodgson. 2007. Application of a real-time underwater surveillance camera in monitoring of fish assemblages on a shallow coral communities in a marine park. In *OCEANS 2007*. IEEE, 1–7.

[35] Minjung Lee, Hyungtae Kim, and Joonki Paik. 2019. Correction of barrel distortion in fisheye lens images using image-based estimation of distortion parameters. *IEEE Access* 7 (2019), 45723–45733.

[36] Zheng Liang, Yafei Wang, Xueyan Ding, Zetian Mi, and Xianping Fu. 2021. Single underwater image enhancement by attenuation map guided color correction and detail preserved dehazing. *Neurocomputing* 425 (2021), 160–172.

[37] Yang Liu, Zhenyue Qin, Saeed Anwar, Pan Ji, Dongwoo Kim, Sabrina Caldwell, and Tom Gedeon. 2021. Invertible denoising network: A light solution for real noise removal. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition.* 13365–13374.

[38] ST Microelectronics. [n. d.]. STM32U535/U545. https://www.st.com/en/microcontrollers-microprocessors/stm32u535-545.html. Accessed: 2024-03-12.

[39] Moodstocks. [n. d.]. JPEC - A Lightweight JPEG encoder in C. https://github.com/Moodstocks/jpec. Accessed: 2024-03-12, Commit: 18fc521c57ca8ede35d3fa0600d6e10aa83c7c85.

[40] Camilo Mora, Derek P. Tittensor, Sina Adl, Alastair G. B. Simpson, and Boris Worm. 2011. How Many Species Are There on Earth and in the Ocean? *PLOS Biology* 9, 8 (08 2011), 1–8. https://doi.org/10.1371/journal.pbio.1001127

[41] Saman Naderiparizi, Mehrdad Hessar, Vamsi Talla, Shyamnath Gollakota, and Joshua R Smith. 2018. Towards {Battery-Free}{HD} video streaming. In *15th USENIX Symposium on Networked Systems Design and Implementation (NSDI 18).* 233–247.

[42] Saman Naderiparizi, Mehrdad Hessar, Vamsi Talla, Shyamnath Gollakota, and Joshua R Smith. 2018. Towards {Battery-Free}{HD} video streaming. In *15th USENIX Symposium on Networked Systems Design and Implementation (NSDI 18).* 233–247.

[43] Saman Naderiparizi, Aaron N Parks, Zerina Kapetanovic, Benjamin Ransford, and Joshua R Smith. 2015. Wispcam: A battery-free rfid camera. In *RFID (RFID), 2015 IEEE International Conference on.* IEEE, 166–173.

[44] Ivan Nagelkerken, Bayden D Russell, Bronwyn M Gillanders, and Sean D Connell. 2016. Ocean acidification alters fish populations indirectly through habitat modification. *Nature Climate Change* 6, 1 (2016), 89–93.

[45] Octopart. [n. d.]. *640H x 480V, Ultra Low-Power,CMOS Digital Image Sensor.* https://datasheet.octopart.com/MT9V011P11STC%3AB-Aptina-datasheet-14184473.pdf

[46] OmniVision. [n. d.]. *Color CMOS VGA (640x480) CameraCubeChip® with OmniPixel®3-HS Technology.* https://www.ovt.com/products/ovm7692/

[47] Danny Pascale. 2006. RGB coordinates of the Macbeth ColorChecker. *The BabelColor Company* 6 (2006), 6.

[48] Harpreet S Sawhney and Rakesh Kumar. 1999. True multi-image alignment and its application to mosaicing and lens distortion correction. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 21, 3 (1999), 235–243.

[49] Faranak Shamsafar, Samuel Woerz, Rafia Rahim, and Andreas Zell. 2022. Mobilestereonet: Towards lightweight deep networks for stereo matching. In *Proceedings of the ieee/cvf winter conference on applications of computer vision.* 2417–2426.

[50] Xi Shen, François Darmon, Alexei A Efros, and Mathieu Aubry. 2020. Ransac-flow: generic two-stage image alignment. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part IV 16.* Springer, 618–637.

[51] Xi Shen, François Darmon, Alexei A Efros, and Mathieu Aubry. 2020. Ransac-flow: generic two-stage image alignment. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part IV 16.* Springer, 618–637.

[52] Infineon Technologies. [n. d.]. Dual N-Channel Small Signal 20 V MOSFET in SOT363 package. https://www.infineon.com/cms/en/product/power/mosfet/small-signal-small-power/bsd840n/. Accessed: 2024-03-12.

[53] Tektronix. [n. d.]. Keithley DMM6500. https://www.tek.com/en/products/keithley/digital-multimeter/dmm6500. Accessed: 2024-03-13.

[54] Radu Timofte, Rasmus Rothe, and Luc Van Gool. 2016. Seven ways to improve example-based single image super resolution. In *Proceedings of the IEEE conference on computer vision and pattern recognition.* 1865–1873.

[55] Freeman W Torralba A, Isola P. 2024. *Foundations of Computer Vision.* The MIT Press.

[56] Fabio Tosi, Alessio Tonioni, Daniele De Gregorio, and Matteo Poggi. 2023. Nerf-supervised deep stereo. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition.* 855–866.

[57] Yosuke Ueki and Masaaki Ikehara. 2021. Underwater image enhancement with multi-scale residual attention network. In *2021 International Conference on Visual Communications and Image Processing (VCIP).* IEEE, 1–5.

[58] Bandhav Veluri, Collin Pernu, Ali Saffari, Joshua Smith, Michael Taylor, and Shyamnath Gollakota. 2023. NeuriCam: Key-Frame Video Super-Resolution and Colorization for IoT Cameras. In *Proceedings of the 29th Annual International Conference on Mobile Computing and Networking.* 1–17.

[59] Hongliang Wang, Wenyu Cai, Junyi Yang, and Qi Chen. 2015. Design of HD video surveillance system for deep-sea biological exploration. In *2015 IEEE 16th International Conference on Communication Technology (ICCT).* IEEE, 908–911.

[60] Kresimir Williams, Alex De Robertis, Zachary Berkowitz, Chris Rooper, and Rick Towler. 2014. An underwater stereo-camera trap. *Methods in Oceanography* 11 (2014), 1–12.

[61] C Winnewisser, F Lewen, and H Helm. 1998. Transmission characteristics of dichroic filters measured by THz time-domain spectroscopy. *Applied Physics A: Materials Science & Processing* 66, 6 (1998).

[62] Yanze Wu, Xintao Wang, Yu Li, Honglun Zhang, Xun Zhao, and Ying Shan. 2021. Towards vivid and diverse image colorization with generative color prior. In *Proceedings of the IEEE/CVF international conference on computer vision.* 14377–14386.

[63] X. 2021. Tidal: Protecting the ocean while feeding humanity sustainably. https://x.company/projects/tidal/.

[64] Tidal X. 2021. Tidal Protecting the ocean while feeding humanity sustainably. https://x.company/projects/tidal/.

[65] Lihe Yang, Bingyi Kang, Zilong Huang, Xiaogang Xu, Jiashi Feng, and Hengshuang Zhao. 2024. Depth Anything: Unleashing the Power of Large-Scale Unlabeled Data. In *CVPR.*

[66] Qiaoyin Yang, Xu A Zhang, Abhijeet Bagal, Wei Guo, and Chih-Hao Chang. 2013. Antireflection effects at nanostructured material interfaces and the suppression of thin-film interference. *Nanotechnology* 24, 23 (2013), 235202.

[67] Shu-Wei Yang, Kuang-Lung Huang, Chien-Yue Chen, and Rong-Seng Chang. 2014. Wide-angle lens design. In *Optical Fabrication and Testing.* Optica Publishing Group, JTu5A–27.

[68] Richard Zhang, Phillip Isola, and Alexei A Efros. 2016. Colorful image colorization. In *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11-14, 2016, Proceedings, Part III 14.* Springer, 649–666.

[69] Richard Zhang, Phillip Isola, and Alexei A Efros. 2016. Colorful image colorization. In *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11-14, 2016, Proceedings, Part III 14.* Springer, 649–666.