Final Year Continual Assessment Project Report
FYP CA Project Report

Underwater Real-Time Object Recognition and Tracking for Autonomous Underwater Vehicle

By

Tan Soon Jin

Department of Computer Science

School of Computing

National University of Singapore

2016/17

Final Year Project Continual Assessment Project Report

Underwater Real-Time Object Recognition and Tracking for Autonomous Underwater Vehicle

By

Tan Soon Jin

Department of Computer Science

School of Computing

National University of Singapore

2016/17

Project No: H021400
Advisor: Prof. Terrence Sim Mong Cheng
Deliverables:
    Report: 1 Volume

Abstract

This project proposes and implements a near real-time vision processing framework on the *Bumblebee AUV (Autonomous Underwater Vehicle)* that participates in Robosub, an international autonomous robotics submarines held annually in San Diego hosted by AUVSI (Association for Unmanned Vehicle Systems International).

The implemented vision system will be deployed on the AUV to complete a series of visual tasks during the competition that mimics real world underwater application such as collecting data on marine life-forms, repairing underwater pipeline etc. Though there are many state-of-the-art vision algorithms developed by the community, the underwater domain poses an entirely different set of challenges such as low contrast, color degradation and underwater perturbations that demands a different vision processing approach.

The primary contribution of the project is to implement a vision framework consisting of a set of modular vision modules and pipelines for real-time object tracking in different underwater conditions. It is essential that the the vision framework is both a) adaptive, b) robust and c) easy to use. With the implemented vision modules, the project's secondary contribution aims to automate parameter selection and model selection; taking the human out of the loop.

Subject Descriptors:
Computer System Implementation
Visual Computing

Implementation Software and Hardware:
Python, ROS (Robot Operating System)

Acknowledgement

I would like to acknowledge the advice and guidance by my supervisor Prof. Terrence Sim Mong Cheng in this project.

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Background on Robosub

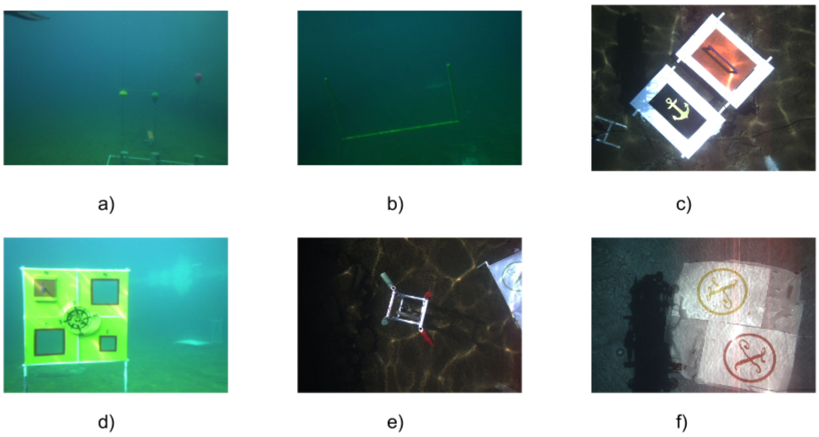
1.1.1 Information about the competition

Robosub is an international AUV competition where students from around the world build their own customized AUV to complete a series of underwater missions that involve both visual tasks and acoustics task. The competition is held annually in TRANSDEC (Transducer Evaluation Center) man-made pool.

Figure 1.1: Aerial view of TRANSDEC. Operational depth of 16 ft for most vision tasks

1.1.2 Description of vision tasks

Vision tasks in Robosub can divided into forward-facing tasks and bottom-facing tasks which poses different sets of challenges. Since the tasks do not vary significantly every year, we can use datasets collected from this year's competition as testbed for our vision algorithms.



a) b) c) d) e) f)

Figure 1.2: Robosub 2016 Vision Tasks. a) Scuttle Ship b) Navigate Channel c) Weigh Anchor d) Set Course e) Bury Treasure (Coins) f) Bury Treasure (Island)

1. **Scuttle Ship (Buoy)** A recurring task where the AUV has identify the correct color buoy and touch it. There are two major challenges with this task:
    (a) Red buoy tends to exhibit color distortion as red wavelength attenuates the fastest (Galdran, Pardo, Picón, & Alvarez-Gila, 2015).
    (b) Non-uniform illumination on top-half of buoys make it hard to distinguish the buoys.
2. **Navigate Channel**

    The AUV is required to move in between and over the PVC pipes.
3. **Weight Anchor**

    Classic object classification task where the AUV is required to drop a marker into the correct bin to obtain maximum points after removing the cover using a manipulator.

4. **Set Course**

   Identification of covered square (orange panel) and remove it. Fire two markers over 2 smaller holes. As yellow and orange are really close on the colour spectrum, this forces us to use other visual cues such as edge for better detection.

5. **Bury Treasure**

   For this task, one has to identify the small cylinders (red and green) and drop them onto their respective colored circles (on the Island). Identifying and distinguishing small objects afar (4 m) underwater is the biggest challenge in this task. Besides that, the dropped cylinders may potentially occlude the circles.

1.2 Challenges in Underwater Image Processing

Many literature such as M, Abhilash, and Supriya (2016) that investigates various underwater image restoration methods cite haze formation which happens as light propagated from object undergoes attenuation and scattering causing image with low contrast. In addition, Beer-Lambert law (Gevers, Gijsenij, Van de Weijer, & Geusebroek, 2012) states relates attenuation of light to properties of water medium; therefore, light components with low wavelength; green and blue are not as easily absorbed compared to red wavelength. This causes underwater images tend to have greenish or bluish color cast.

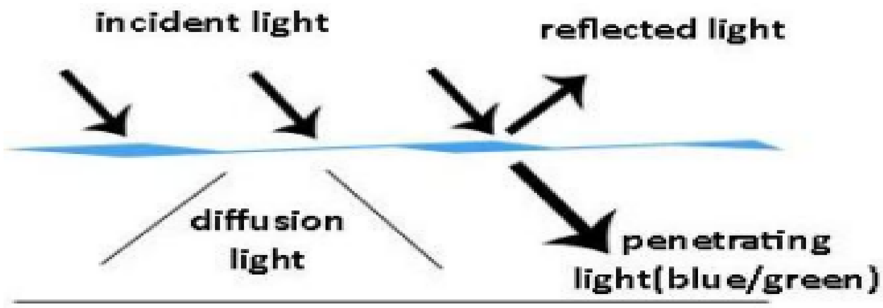


Figure 1.3: Absorption of light at the surface

1.3 Project Requirements Analysis

Though it is the objective of the project to design a vision framework for the Robosub missions, the vision framework should also be easily extended to work for more complex real world applications.

1.3.1 Nature of tasks

1. Vision algorithms perform with acceptable accuracy under the following conditions:

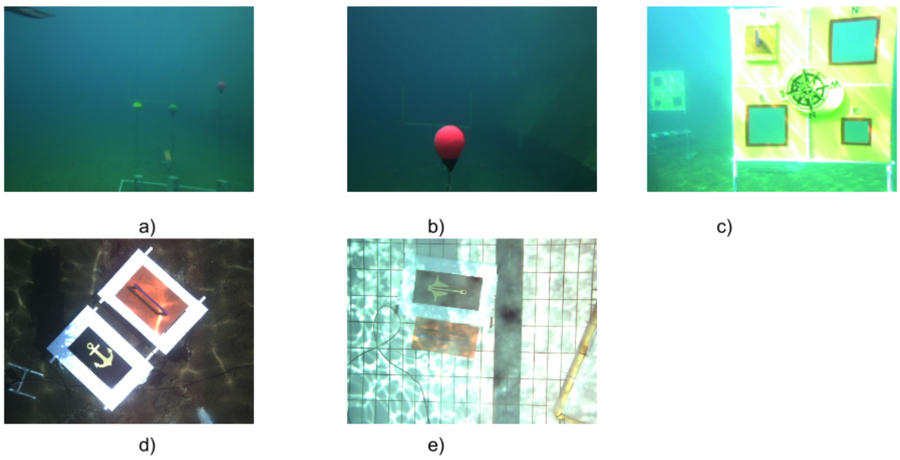


Figure 1.4: Different vision challenges. a) Haze formation b) Partial occlusion c) Non-uniform illumination d) Sunlight flickers e) Shadow

2. Low detection latency (near real-time)

   AUV needs to make swift decision based on sensor inputs to complete task under time constraints (same for real world time critical mission i.e underwater mine detection)

3. Geometric properties of objects are made known in advance

4. Short-period single target tracking for task (unlike video surveillance application)

5. Able to detect objects from far away (5m) and near distance (for manipulation task)

Chapter 2

Literature Review

This review is conducted with the purpose to investigate and select most suitable algorithms that generate the best result on the Robosub datasets. Since every teams who participate in Robosub are required to submit a journal paper,vision algorithms deployed by top-peforming schools such as Cornell University, University of Florida and École de technologie supérieure provide valuable insights on image processing that are effective in underwater environment. Besides that, review of popular image processing techniques in particular on topics like object detection, object tracking, color constancy, saliency mechanism, detection proposals and adapatation of algorithms.

2.1 Preprocessing

2.1.1 Underwater Image Enhancement

The paper by Garcia, Nicosevici, and Cufí (2002) compared methods such as homomorphic filtering and local adaptive histogram equalization (Contrast Limited Adaptive Histogram) which considers that image is a product of illumination and reflectance properties. However, homomorphic filter has the benefit of preserving sharp edges while attenuating non-uniform illumination. On the other hand, by only redistributing pixels exceeding a clipping level to increase contrast of an image, CLAHE manages to reduce noise amplification in normal local histogram equalization.

Instead of relying on a single image, Gracias, Negahdaripour, Neumann, Prados, and Garcia (2008) recover corrupted underwater image by finding the difference between the current frame with temporal median of a registered set of N frames. Image dehazing is equally as important to ensure good performance of further image processing operation such feature detection. Kaiming, Jian, and Xiaoou (2011) proposed a single image dehazing method using the dark channel prior which states that haze-free image contains local region with low intensities in at least one color channel. Galdran et al. (2015) propose a variant of dark channel prior for underwater environment, the Red Channel method as red color shows most degradation in turbid water medium. From another perspective, Ancuti and Bekaert (2011) takes a fusion-approach to recover the original image by generating a few weight maps that correlates with intrinsic properties of the image itself. A color corrected and contrast enhanced of the input image are used to generate different weight maps that are fused using a Laplacian multi-scale strategy to generate a smoothed output image. This method has the benefit of using a single image but the weight maps must be combined with different weightage to achieve an ideal result.

2.1.2 Color Constancy

Color cue plays an important role to distinguish different objects such as the small cylinders in Robosub that requires sorting by color. The ability to account for color of the light source is called color constancy. The work of Gijsenij, Gevers, and Van De Weijer (2011) analyzes various color constancy algorithms. Attention is paid especially on low-level statistics methods that are computationally inexpensive compared to learning-based methods. The Grey-World (Buchsbaum, 1980) estimate the color of the light source by estimating the average color in the image assuming that any deviation from average color (Grey) is caused by illuminants. The White-Patch method (Land & others, 1977) estimates the color of light source by computing the maximum response in individual RGB color channels. Finlayson and Trezzi (2004) shows that both Grey-World and White-Patch algorithms are special instantiation of a more general color constancy algorithm based on Minkowski norm called Shades of Grey. Their investigation of best illumination estimation suggests using Minkowski norm, p = 6 to obtain optimal performance.

Though we see new method such as the Color Rabbit (Bani?? & Lon??ari??, 2014) which combine multiple local illumination estimations to a global one, these class of methods are more computationally expensive which is not suitable for real-time application. Inspired by primary visual cortex (V1) of human visual system (HVS), Gao, Yang, Li, and Li (2013) estimate the true illuminant color of a scene by computing the maximum response in separate RGB channels of the responses of double-opponent cells. This method is shown to perform better on outdoor scenes from Gehler-Shi dataset where the mean reflectance is not achromatic which is assumed by Grey-World based methods.

2.2 Saliency Region Detection

Ability of human visual system (HVS) to selectively process only the salient visual stimuli, specifically salient object detection helps to reduce computation time of object recognition that traditionally relies of sliding-window approach to detect object of interest. Achanta, Hemami, Estrada, and Susstrunk (2009) estimate centre-surround contrast using color and luminance features using a frequency-tuned approach to generate high-resolution saliency map. In contrast, biological inspired method of (Itti, Koch, & Niebur, 1998) that computes centre-surround contrast using Difference of Gaussian (DoG) which generates low resolution map and ill-defined boundaries because of down sampling of original image.Because saliency detection often work poorly in low contrast environment i.e underwater environment, work of Van De Weijer and Gevers (2005) boost local color information by analyzing isosalient colour derivatives. Cao and Cheikh (2010) extended work of Van de Weijer as Gaussian derivatives of each opponent color to get a better iso-salient transformation.

2.3 Detection Proposals

Relying on saliency mechanism is insufficient in perturbed underwater condition; therefore, different detection proposals algorithms are investigated. Hosang, Benenson, Dollár, and Schiele (2015) cited that "detection proposals" which can be grouped into a) grouping proposal methods and b) Window scoring proposals methods are used extensively by top performing object detectors in PASCAL and ImageNet. On top of reduced computation cost by avoiding exhaustive sliding window approach, detection proposals improve recall by filtering out false positives. Recent work of Winschel, Lienhart, and Eggert (2016) combines top performing detection proposals methods, SelectiveSearch (Uijlings, van de Sande, Gevers, & Smeulders, 2013) and EdgeBox (Zitnick & Dollár, 2014). Though detection proposals allow for faster object recognition, it is important that it does not filter out object of interest and incur more computation costs that out weights time saved.

2.4 Object Detection and Tracking

An overall review of journal papers submitted by top-performing teams in Robosub shows a general trend of combining surprisingly simple computer vision techniques such as adaptive color thresholding, edge detection i.e Canny Edge (Canny, 1986), and contour analysis i.e Hu moment (Hu, 1962). Team CUAUV (Cornell AUV) proposes adaptive color thresholding on different color spaces such as LAB, LUV and YCrCb where the individual masks are combined to form final binarized mask. This is a blob-based detection approach where contour generated by OpenCV's implementation of (Suzuki & others, 1985) will be matched against known geometric properties of desired object of interest. Walters, Sauder, Nezvadovitz, Voight, Gray, Schwartz, and Walters, P and Sauder, N and Nezvadovitz, J and Voight, F and Gray, A and Schwartz (2014) use particle filter approach to detect and track object of interest. Known for its ability to deal with non-linear noise and multi-modal hypotheses (Isard & Blake, 1998), particle filter has the ability to recover from wrongly tracked objects. Though more sophisticated techniques such as neural-network classification is deployed, teams still generally rely on low-level visual cues such as color and edge. This may be attributed to simplicity and efficiency of mentioned algorithms. Benoit, Goulet, Bouchard-d'Haese, Bouzidi, Carrier, Couturier, Desjardins, Dozois, Fortier, Langlois, Ritchie, and Prévost (2014) focuses on developing sophisticated vision tuning client that allows for rapid prototyping via "mix and match" approach to design a suitable vision pipeline for each individual vision tasks.

Chapter 3

Design & Implementation

3.1 Vision Architecture

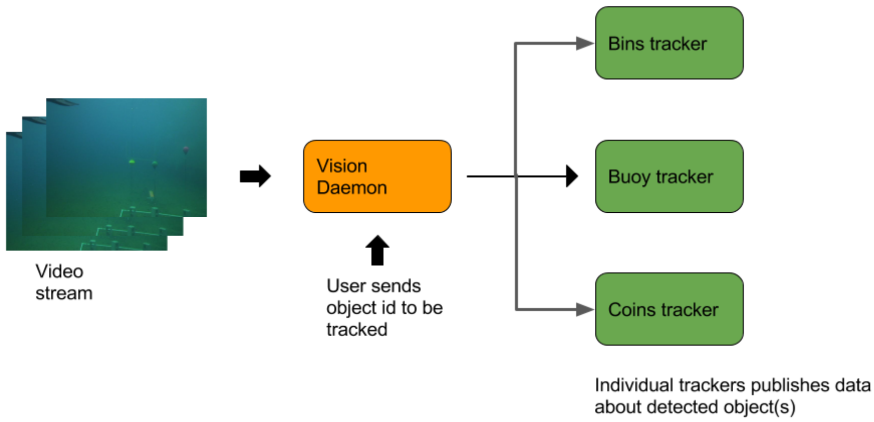


Figure 3.1: BBAUV Vision Architecture

Our vision framework is implemented as a vision daemon running in the background when the AUV is launched. The vision framework is integrated with ROS (Robotics Operating System) (Quigley, Conley, Gerkey, Faust, Foote, Leibs, Wheeler, & Ng, 2009), an open source meta-operating systems used widely in robotics which provides message interface for inter-process communication (IPC). The vision daemon will initialized a new tracker upon request from the user (if the object is not currently being tracked). One advantage of this approach allows for parallel development of vision algorithms before competition which not speed up development but resulting in modular vision algorithms that are substitutable.

3.2 Methodology

Individual object trackers are implemented using a tracking-by-detection approach where theese trackers had already been implemented and deployed in Robosub 2016. Below shows the vision pipeline of the baseline object tracker:

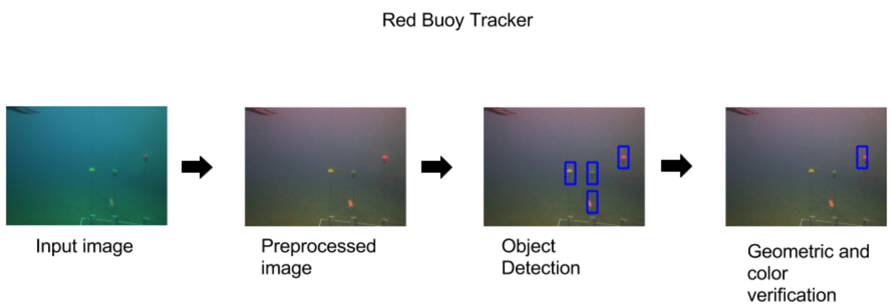


Figure 3.2: Object Tracking Pipeline

3.2.1 Preprocessing

Input image first undergoes preprocessing before being processed by individual object detector. A design decision has been made to perform preprocessing on demand to a) reduce computation cost and b) customize preprocessing for different type of vision tasks. From past experiences, objects with distinct color from environment and bottom-facing objects can be detected without much preprocessing. Preprocessing can be separated into color constancy algorithms and underwater image enhancement algorithms.

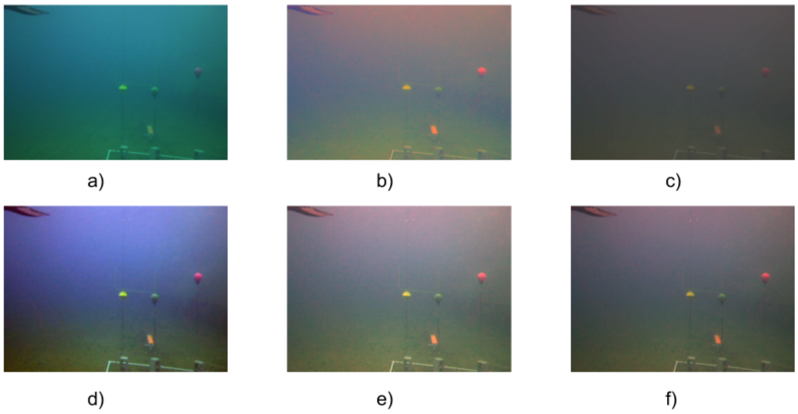


Figure 3.3: Color constancy algorithms a) Original image b) Finlayson's comprehensive normalization c) Grey world d) Image Adaptive Contrast Enhancement (IACE) e) Non-iterative normalization f) Shade of Grey

Implemented Color Constancy Algorithms

Various color constancy yields different results based on different underwater conditions. We can observe that some color constancy algorithms produce image with red hue which may affect performance of color-based object detectors. It must be known that gamma correction is performed after performing Grey-World algorithm to produce image with sufficient lighting. Gijsenij et al. (2011) suggests that several approaches are combined to achieve optimal result.

Underwater Image Enhancement

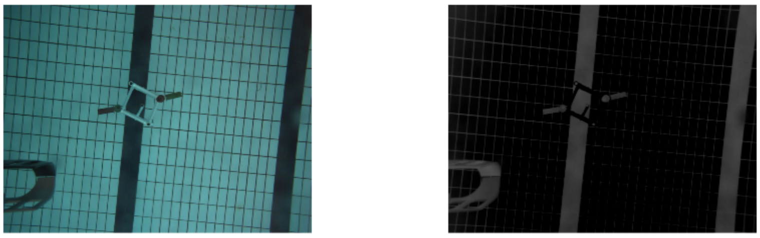
1. Gamma correction

   To reduce effect of overexposure or underexposure because of camera settings. Sudden change in illumination because of cloud movement or position of the sun can be catastrophic.

2. Homomorphic filter

   To reduce flickering effect of bottom- facing tasks. This is implemented on the spatial domain (Nnolim & Lee, 2008)

3.3 Saliency Region Detection



a) b)

Figure 3.4: Salient object detection of cylinders a) original image where color of cylinders severely degraded b) saliency map still mangages to capture the cylinders

The saliency detection approach of (Achanta et al., 2009) is implemented to detect salient objects underwater as competition obstacles are often more salient than other features. Experiments have shown that images needed to be contrast enhanced and color corrected to achieve reliable detection.

Chapter 4

Plan for the next semester

4.1 Implement more robust object detection and object tracking algorithms

4.1.1 Improve object detection

Moving on, achieving consistent and accurate detection of objects in more difficult environments such as partial occlusion, sudden illumination change and presence of shadow. Taking inspirations from success of top object detector such as Yang, Yan, Lei, and Li (2014) that aggregate multiple-channel features; a multi-cues approach that includes both global and local features will be implemented to increase robustness of current object detector.

4.1.2 Better object tracking

One limitation of current tracking approach is neglecting prior tracked position which can help to prevent drift and reduce false positives. Instead of traditional particle filter that ignores current measurement in its system model and use only color model for its observation model, the project look to integrate more features such as optical flow and salient features.

4.2 Automate parameters selection and model selection

Unlike approach taken by other schools that rely on manual visual tuning to achieve robust object tracking, the project aims to take inspirations from works of (Zhang, Zhu, & Roy-Chowdhury, 2016) and (Collins, Liu, & Leordeanu, 2005) that attempt to map algorithms-parameters pair to specific dataset. A similarity function is then used to measure difference between test images with trained images.

4.3 Experimental Setup

In order to perform evaluation on proposed vision framework, several performance metrics will be used but the project adopts approach of Luo, Xing, Zhang, Zhao, and Kim (2014) that uses: a) MOTA (Multiple Object Tracking Accuracy) b) MOTP (Multiple Object Tracking Precision) Average Overlap (Intersection-over-Union of bounding box).

4.3.1 Real-world dataset

Recorded images from Robosub 2015, Robosub 2016 and Queenstown Pooltest will be labelled and used to evaluate performance of proposed vision algorithms. These datasets will be divided according to different challenges such as buoy detection, bins detection and coins detection.

4.4 Proposed Time-line

Time Period	Work to be done
December Holiday	Propose and implement a set of object detection and object tracking that will work in different underwater conditions
January	Validate proposed approach by comparing their performance with the baseline. Identify ways to automate parameter selection and model selection based for different tasks or scenarios
February, March, April	Report writing on experimental results and findings

Chapter 5

Object Tracking

5.1 Current benchmarks result

5.1.1 PAMI 15: Object Tracking Benchmark

(Wu, Lim, & Yang, 2015)

1. **Representation Scheme**

   (a) Eigenspace basis functions (Black & Jepson, 1998)

   (b) Sparse representation (Zhong, Lu, & Yang, 2012)

   (c) Sparse templates and outlier detection (Mei & Ling, 2011)

   (d) Local sensitivity histogram that captures spatial contribution of each pixels (He, Yang, Lau, Wang, & Yang, 2013)

   (e) Covariance descriptor (Tuzel, Porikli, & Meer, 2006)

   (f) Online Structured Output SVM (Hare, Golodetz, Saffari, Vineet, Cheng, Hicks, & Torr, 2016)

2. **Search Mechanism**

   (a) Particle filter with effective observation model

   (b) Distribution Fields (Sevilla-Lara & Learned-Miller, 2012)

3. **Model Update: Account for appearance variation**
   (a) Online AdaBoost (Grabner, Grabner, & Bischof, 2006)
4. **Context & Fusion of Trackers**
   (a) Mining auxiliary objects for verification (Yang, Wu, & Hua, 2009)
   (b) Identify distractors and supporters as random forest (Dinh, Vo, & Medioni, 2011)
   (c) Choosing best trackers based on MCMC (Kwon & Lee, 2011)
   (d) Fallback models to handle different situation (Santner, Leistner, Saffari, Pock, & Bischof, 2010)
5. **Evaluation methodology**
   (a) Precision plot: percentage of frames in which the estimated locations are within a given threshold distance of the ground-truth positions (20 pixels euclidean distance).
   (b) Success plot: intersection over union (average overlap score) with t = 0.5
   (c) AUC: Area Under Curve of success plots
   (d) TRE (Temporal robustness evaluation): average result over different initial frame
   (e) SPE (Spatial robustness evaluation): shifting or scaling grund truth bounding box
   (f) Evaluate with restart
6. **Conclusion**
   (a) STRUCK (Hare et al., 2016)
   (b) SCM (Zhong et al., 2012)
   (c) ASLA (Jia, Lu, & Yang, 2012)
   (d) CSK (Henriques, Caseiro, Martins, & Batista, 2012)
   (e) L1APG (Bao, Wu, Ling, & Ji, 2012)
   (f) OAB (Grabner et al., 2006)

5.1.2 VOT 2015

(Kristan, Matas, Leonardis, Felsberg, Cehovin, Fernandez, Vojir, Hager, Nebehay, & Pflugfelder, 2015)

1. **Performanc measure**
   (a) Accuracy: average overlap score
   (b) Robustness: average failure rate
2. **Top-performing trackers**
   (a) MDNet (Nam & Han, 2015)
   (b) SRDCF (Danelljan, Hager, Shahbaz Khan, & Felsberg, 2015)
   (c) EBT (Zhu, Porikli, & Li, 2015)
   (d) sPST (Hua, Alahari, & Schmid, 2015)
   (e) LDP (Lukežič, Čehovin, & Kristan, 2016)

5.1.3 MOT 2016

(Milan, Leal-Taixe, Reid, Roth, & Schindler, 2016)

1. **Performancre measures**
   (a) MOTA (Multiple Object Tracking Accuracy)

$$MOTA = \sum_t \frac{FN_t + FP_t + IDSW_t}{\sum_t GT}$$

   IDSW Number of mismatch target

   (b) MOTP (Multiple Object Tracking Precision) measures localization precision

$$MOTP = \frac{\sum_{t,i} d_{t,i}}{\sum_t c_t}$$

   c_t number of matches in frame t

   $d_{t,i}$ bounding box overlap

(c) Track quality measurement

Mostly Tracked Detected 80% of the time

Partially tracked Detected 20% of the time

2. **Accuracy ranking**

(a) NOMT (Choi, 2015)

(b) Online Strong and Weak PHT-PF (Sanchez-Matilla, Poiesi, & Cavallaro, 2016)

(c) DP NMS (Pirsiavash, Ramanan, & Fowlkes, 2011)

Chapter 6

Automatic Machine Learning

6.1 Adaptive Algorithm and Platform Selection for Visual Detection and Tracking

6.1.1 Main Ideas

Suggests the best algorithm-parameter pair given characteristics of the video. Separates into two phases:

1. Design Phase: Learn mapping between scenario in training dataset with algorithm-parameter pair

2. Runtime Phase: Calculate similarity between test video and training dataset to choose most similar dataset in database

6.1.2 Similarity measures

1. $S(T_i, R_j) = \epsilon^{-d(T_i, R_j)}$

   Feature-distance is calculated by the geodesic flow. Projecting feature onto Grasmann manifold.

$$t_i^T W_{ij} r_j = \int_0^1 (\theta(y)t_i)^T (\theta(y) r_j) dy \tag{6.1}$$

$$d(T_i, R_j) = t_i^T W_{ij} t_i + r_j^T W_{ij} r_j - 2t_i^T W_{ij} r_j \tag{6.2}$$

6.2 On-line selection of discriminative features

(Collins et al., 2005)

6.2.1 Feature Selection

Mainly consists of: a) feature selection criterion and b) search strategy. Evaluate each feature based on *class separability* from surrounding.

1. Estimate histogram of detected object and surrounding (center-surround principle)

2. Calculate log-likelihood ratio of these distributions

$$L(i) = \log \frac{\max p(i), 0.0001}{\max q(i), 0.0001}$$

3. Calculate variance ratio of log-likehood

$$VR(L,p,q) = \frac{var(L; \frac{p+q}{2})}{var(L,p) + var(L,q)}$$

6.3 Automatic Parameter Adapation for MOT

(Chau, Thonnat, & Brémond, 2013)

6.3.1 Offline Learning

Contextual Feature Extraction

1. Density of objects

2. Occlusion level

3. Contrast of object

4. Contrast variance between objects

5. Area of objects

6. Area Variance between objects

Context Code book

Divides training video into chunks with similar context code book

1. Code book consists of a set of code words
2. Code word

$$cw^k = meanvalue, maximum, minimum, frequencies$$

3. Context distance is ratio between mean value of code word

6.3.2 Online steps

Context detection

Given an input video stream, if context of video is in database do nothing, else activate parameter tuning

Parameter tuning

Chooses context that is closest from trained database

6.4 Efficient and Robust Automated Machine Learning (winner of AutoML)

(Feurer, Klein, Eggensperger, Springenberg, Blum, & Hutter, 2015a)

6.4.1 Main Ideas

Extends from Auto-Weka which uses tree-based Bayesian Optimization (Thornton, Hutter, Hoos, & Leyton-Brown, 2013) to model relationship between hyperparameters setting and performance. A random forest based SMAC approach is used to solve CASH problem.

6.4.2 Meta-learning

Selects instatiations of machine learning frameworks as seed for Bayesian Optimization (seeding). Train hyperparameters that perform best on offline data using Bayesian Optimization.

6.4.3 Automated ensemble construction of models

Uses ensemble selection to greedily add models learned from Bayesian Optimization to prevent over-fitting

6.4.4 Practical application

(Feurer, Klein, Eggensperger, Springenberg, Blum, & Hutter, 2015b) Based on auto-sklearn components . Alternatives include (Caruana, Niculescu-Mizil, Crew, & Ksikes, 2004):

1. Hyperopt-sklearn (Komer, Bergstra, & Eliasmith, 2014)

2. Auto-Weka (Thornton et al., 2013)

6.5 A brief review of ChaLearn AutoML Challenge

(Guyon, Chaabane, Escalante, Escalera, Jajetic, Lloyd, Macia, Ray, Romaszko, Sebag, & others, 2016)

6.5.1 Preprocessing

1. Zero mean and unit variance (ZMUV)

2. non-linear transformation (log)

3. dimensionality reduction (PCA, ICA)

6.5.2 Model selection

1. K-fold cross-validation

2. Transferring knowledge from phase to phase

Chapter 7

Next Semester

7.1 Existing problems

1. Very costly to acquire large training data
2. Manual selection of algorithms and parameters based on expert knowledge required
3. Expensive sensors needed to achieve robust tracking. Non-visual based sensors i.e LIDAR and sonar provides competitive edge
4. State-of-the art tracking algorithm has high detection latency. Not suitable for real-time application

7.2 Objectives

1. Robustness
   - Different water bodies i.e lake, ocean
   - Changes in illuminations
   - Shadow
2. Automation
   - Automatically chooses algorithm and parameters for preprocessing offline

- Automatically tune camera parameter to achieve optimal image for further processing
- Recommends object detection algorithm given image

3. Flexibility

   - Easily adapted to different Robosub missions
   - Algorithm scales with more training data

4. Rapid Prototyping

   Very fast iteration with limited data collection. Able to learn on the fly or suggest what data to collect

7.3 How do we measure success ?

1. Accuracy across different challenging dataset
2. Precision of bounding box
3. FPS
4. Number of training data vs accuracy

7.4 Mental blocks

1. Ensemble approach vs automatically choosing the best algorithm
2. Focus on preprocessing vs working on color space that is not affected by illuminations and color degradation
3. Domain adaptation / transfer learning vs independent task learning
4. Using multiple cues and features vs simple features that just work
5. One general framework that works for different tasks while allowing unique strategy to be built for specific task
6. How much invariance needed to be enforced ? Affine ?

7.5 Innovations

1. Non-parametric approach

2. Ensemble approach i.e bagging , boosting

3. Detection free approach when near object

4. Transfer Learning

5. Combining local and global features

6. Automatic mission testing

7. Using visual prior to encode belief

8. Tool for rapid annotation on the fly due to competition need

9. Synthesize data to test robustness and increase training dataset

7.6 Methodology

7.6.1 Automatic camera tuning

1. Make use of image entropy to tune parameters

2. Using mean intensity of brightness

7.6.2 Choosing color space

1. Illumination-invariant color space

2. Work on chromacity of the objects

3. Different color space for different task

7.6.3 Image enhancement

1. Contrast stretching
2. Gamma correction
3. Image fusion
4. Histogram Equalization
5. Removing flicker

7.6.4 Object Proposals

Focus on identifying all potential objects in the scene

1. EdgeBox
2. SelectiveSearch
3. Saliency
4. MSER
5. BING
6. Background Subtraction

7.6.5 Object Detection

1. HOG
2. Integral Channel Feature
3. Color Histogram
4. Correlation Filter
5. HuMoment
6. GIST

7. Binary Feature Detectors

8. Haar like feature

7.6.6 Object Tracking

1. Particle Filter

2. Mean Shift

3. Covariance tracking

References

Achanta, R., Hemami, S., Estrada, F., & Susstrunk, S. (2009). Frequency-tuned salient region detection. *Computer vision and pattern recognition, 2009. cvpr 2009. ieee conference on* (pp. 1597–1604), IEEE, 2009.

Ancuti, C., & Bekaert, P. (2011). Enhancing Underwater Images by Fusion. *Computer*, , 2011, 4503–4503.

Bani??, N., & Lon??ari??, S. (2014). Color rabbit: Guiding the distance of local maximums in illumination estimation. *International Conference on Digital Signal Processing, DSP, 2014-Janua*(Cd), 2014, 345–350.

Bao, C., Wu, Y., Ling, H., & Ji, H. (2012). Real time robust l1 tracker using accelerated proximal gradient approach. *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on* (pp. 1830–1837), IEEE, 2012.

Benoit, M., Goulet, M.-A. B., Bouchard-d'Haese, F., Bouzidi, R., Carrier, J., Couturier, É., Desjardins, V., Dozois, A., Fortier, M., Langlois, F., Ritchie, K., & Prévost, J. S.-J. (2014). S.O.N.I.A - Autonomous Underwater Vehicle. , 2014.

Black, M. J., & Jepson, A. D. (1998). Eigentracking: Robust matching and tracking of articulated objects using a view-based representation. *International Journal of Computer Vision*, *26*(1), 1998, 63–84.

Buchsbaum, G. (1980). A spatial processor model for object colour perception. *Journal of the Franklin institute*, *310*(1), 1980, 1–26.

Canny, J. (1986). A computational approach to edge detection. *IEEE Transactions on pattern analysis and machine intelligence*, (6), 1986, 679–698.

Cao, G., & Cheikh, F. A. (2010). Salient region detection with opponent color boosting. *2010 2nd European Workshop on Visual Information Processing, EUVIP2010*, , 2010, 13–18.

Caruana, R., Niculescu-Mizil, A., Crew, G., & Ksikes, A. (2004). Ensemble selection from libraries of models. *Proceedings of the twenty-first international conference on Machine learning* (p. 18), ACM, 2004.

Chau, D. P., Thonnat, M., & Brémond, F. (2013). Automatic parameter adaptation for multi-object tracking. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, *7963 LNCS*, 2013, 244–253.

Choi, W. (2015). Near-online multi-target tracking with aggregated local flow descriptor. *Proceedings of the IEEE International Conference on Computer Vision* (pp. 3029–3037), 2015.

Collins, R. T., Liu, Y., & Leordeanu, M. (2005). Online selection of discriminative tracking features. *IEEE transactions on pattern analysis and machine intelligence*, *27*(10), 2005, 1631–1643.

Danelljan, M., Hager, G., Shahbaz Khan, F., & Felsberg, M. (2015). Learning spatially regularized correlation filters for visual tracking. *Proceedings of the IEEE International Conference on Computer Vision* (pp. 4310–4318), 2015.

Dinh, T. B., Vo, N., & Medioni, G. (2011). Context tracker: Exploring supporters and distracters in unconstrained environments. *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on* (pp. 1177–1184), IEEE, 2011.

Feurer, M., Klein, A., Eggensperger, K., Springenberg, J., Blum, M., & Hutter, F. (2015a). Efficient and robust automated machine learning. *Advances in Neural Information Processing Systems* (pp. 2962–2970), 2015.

Feurer, M., Klein, A., Eggensperger, K., Springenberg, J., Blum, M., & Hutter, F. (2015b). Efficient and robust automated machine learning. In C. Cortes, N. D. Lawrence, D. D. Lee, M. Sugiyama, & R. Garnett (Eds.), *Advances in neural information processing systems 28* (pp. 2962–2970). Curran Associates, Inc.

Finlayson, G. D., & Trezzi, E. (2004). Shades of gray and colour constancy. *Color and Imaging Conference*, Vol. 2004 (pp. 37–41), Society for Imaging Science and Technology, 2004.

Galdran, A., Pardo, D., Picón, A., & Alvarez-Gila, A. (2015). Automatic Red-Channel underwater image restoration. *Journal of Visual Communication and Image Representation*, *26*, 2015, 132–145.

Gao, S., Yang, K., Li, C., & Li, Y. (2013). A Color Constancy Model with Double-Opponency Mechanisms. *Computer Vision (ICCV), 2013 IEEE International Conference on*, , 2013, 929–936.

Garcia, R., Nicosevici, T., & Cufí, X. (2002). On the way to solve lighting problems in underwater imaging. *OCEANS'02 MTS/IEEE*, Vol. 2 (pp. 1018–1024), IEEE, 2002.

Gevers, T., Gijsenij, A., Van de Weijer, J., & Geusebroek, J.-M. (2012). *Color in computer vision: fundamentals and applications*, Vol. 23. John Wiley & Sons.

Gijsenij, A., Gevers, T., & Van De Weijer, J. (2011). Computational color constancy: Survey and experiments. *IEEE Transactions on Image Processing*, *20*(9), 2011, 2475–2489.

Grabner, H., Grabner, M., & Bischof, H. (2006). Real-time tracking via on-line boosting. *BMVC*, Vol. 1 (p. 6), 2006.

Gracias, N., Negahdaripour, S., Neumann, L., Prados, R., & Garcia, R. (2008). A motion compensated filtering approach to remove sunlight flicker in shallow water images. *Oceans 2008*, , 2008, 1–7.

Guyon, I., Chaabane, I., Escalante, H. J., Escalera, S., Jajetic, D., Lloyd, J. R., Macia, N., Ray, B., Romaszko, L., Sebag, M., et al. (2016). A brief review of the chalearn automl challenge: Any-time any-dataset learning without human intervention. *International Conference in Machine Learning (ICML 2016) Workshops*, 2016.

Hare, S., Golodetz, S., Saffari, A., Vineet, V., Cheng, M.-M., Hicks, S. L., & Torr, P. H. (2016). Struck: Structured output tracking with kernels. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *38*(10), 2016, 2096–2109.

He, S., Yang, Q., Lau, R. W., Wang, J., & Yang, M.-H. (2013). Visual tracking via locality sensitive histograms. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (pp. 2427–2434), 2013.

Henriques, J. F., Caseiro, R., Martins, P., & Batista, J. (2012). Exploiting the circulant structure of tracking-by-detection with kernels. *European conference on computer vision* (pp. 702–715), Springer, 2012.

Hosang, J., Benenson, R., Dollár, P., & Schiele, B. (2015). What makes for effective detection proposals? *Arxiv*, , 2015, 2014.

Hu, M.-K. (1962). Visual pattern recognition by moment invariants. *IRE transactions on information theory*, *8*(2), 1962, 179–187.

Hua, Y., Alahari, K., & Schmid, C. (2015). Online object tracking with proposal selection. *Proceedings of the IEEE International Conference on Computer Vision* (pp. 3092–3100), 2015.

Isard, M., & Blake, A. (1998). Condensation—conditional density propagation for visual tracking. *International journal of computer vision*, *29*(1), 1998, 5–28.

Itti, L., Koch, C., & Niebur, E. (1998). A model of saliency-based visual attention for rapid scene analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *20*(11), 1998, 1254–1259.

Jia, X., Lu, H., & Yang, M.-H. (2012). Visual tracking via adaptive structural local sparse appearance model. *Computer vision and pattern recognition (CVPR), 2012 IEEE Conference on* (pp. 1822–1829), IEEE, 2012.

Kaiming, H., Jian, S., & Xiaoou, T. (2011). Single Image Haze Removal Using Dark Channel Prior. *IEEE Trans Pattern Anal Mach Intell*, *33*(12), 2011, 2341–2353.

Komer, B., Bergstra, J., & Eliasmith, C. (2014). Hyperopt-sklearn: automatic hyperparameter configuration for scikit-learn. *ICML workshop on AutoML*, 2014.

Kristan, M., Matas, J., Leonardis, A., Felsberg, M., Cehovin, L., Fernandez, G., Vojir, T., Hager, G., Nebehay, G., & Pflugfelder, R. (2015). The visual object tracking vot2015 challenge results. *Proceedings of the IEEE International Conference on Computer Vision Workshops* (pp. 1–23), 2015.

Kwon, J., & Lee, K. M. (2011). Tracking by sampling trackers. *2011 International Conference on Computer Vision* (pp. 1195–1202), IEEE, 2011.

Land, E. H., et al. (1977). *The retinex theory of color vision.* Citeseer.

Lukežič, A., Čehovin, L., & Kristan, M. (2016). Deformable parts correlation filters for robust visual tracking. *arXiv preprint arXiv:1605.03720*, , 2016.

Luo, W., Xing, J., Zhang, X., Zhao, X., & Kim, T.-K. (2014). Multiple Object Tracking: A Literature Review. *arXiv, 1-39. Retrieved from http://arxiv.org/abs/1409.7618*, *V*(212), 2014.

M, A. R. S., Abhilash, S., & Supriya, M. H. (2016). A Comparative Study of Various Methods for Underwater Image Enhancement and Restoration. *6*(2), 2016, 30–33.

Mei, X., & Ling, H. (2011). Robust visual tracking and vehicle classification via sparse representation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *33*(11), 2011, 2259–2272.

Milan, A., Leal-Taixe, L., Reid, I., Roth, S., & Schindler, K. (2016). Mot16: A benchmark for multi-object tracking. *arXiv preprint arXiv:1603.00831*, , 2016.

Nam, H., & Han, B. (2015). Learning multi-domain convolutional neural networks for visual tracking. *arXiv preprint arXiv:1510.07945*, , 2015.

Nnolim, U., & Lee, P. (2008). Homomorphic filtering of colour images using a spatial filter kernel in the hsi colour space. *Instrumentation and Measurement Technology Conference Proceedings, 2008. IMTC 2008. IEEE* (pp. 1738–1743), IEEE, 2008.

Pirsiavash, H., Ramanan, D., & Fowlkes, C. C. (2011). Globally-optimal greedy algorithms for tracking a variable number of objects. *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on* (pp. 1201–1208), IEEE, 2011.

Quigley, M., Conley, K., Gerkey, B., Faust, J., Foote, T., Leibs, J., Wheeler, R., & Ng, A. Y. (2009). Ros: an open-source robot operating system. *ICRA workshop on open source software*, Vol. 3 (p. 5), Kobe, Japan, 2009.

Sanchez-Matilla, R., Poiesi, F., & Cavallaro, A. (2016). Online multi-target tracking with strong and weak detections. *European Conference on Computer Vision* (pp. 84–99), Springer, 2016.

Santner, J., Leistner, C., Saffari, A., Pock, T., & Bischof, H. (2010). Prost: Parallel robust online simple tracking. *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on* (pp. 723–730), IEEE, 2010.

Sevilla-Lara, L., & Learned-Miller, E. (2012). Distribution fields for tracking. *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on* (pp. 1910–1917), IEEE, 2012.

Suzuki, S., et al. (1985). Topological structural analysis of digitized binary images by border following. *Computer Vision, Graphics, and Image Processing*, *30*(1), 1985, 32–46.

Thornton, C., Hutter, F., Hoos, H. H., & Leyton-Brown, K. (2013). Auto-weka: Combined selection and hyperparameter optimization of classification algorithms. *Proceedings of the 19th ACM SIGKDD international conference on Knowledge discovery and data mining* (pp. 847–855), ACM, 2013.

Tuzel, O., Porikli, F., & Meer, P. (2006). Region covariance: A fast descriptor for detection and classification. *European conference on computer vision* (pp. 589–600), Springer, 2006.

Uijlings, J. R., van de Sande, K. E., Gevers, T., & Smeulders, A. W. (2013). Selective search for object recognition. *International journal of computer vision*, *104*(2), 2013, 154–171.

Van De Weijer, J., & Gevers, T. (2005). Boosting saliency in color image features. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, *1*(c), 2005, 365–372.

Walters, P., Sauder, N., Nezvadovitz, J., Voight, F., Gray, A., Schwartz, E. M., & Walters, P and Sauder, N and Nezvadovitz, J and Voight, F and Gray, A and Schwartz, E. (2014). SubjuGator 2014: Design and Implementation of a Modular, Fault Tolerant AUV. *AUVSI Foundation's 17th Annual RoboSub Competition, San Diego, CA*, , 2014, 1–8.

Winschel, A., Lienhart, R., & Eggert, C. (2016). Diversity in Object Proposals. (C), 2016.

Wu, Y., Lim, J., & Yang, M.-H. (2015). Object tracking benchmark. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *37*(9), 2015, 1834–1848.

Yang, B., Yan, J., Lei, Z., & Li, S. Z. (2014). Aggregate channel features for multi-view face detection. *IJCB 2014 - 2014 IEEE/IAPR International Joint Conference on Biometrics*, , 2014.

Yang, M., Wu, Y., & Hua, G. (2009). Context-aware visual tracking. *IEEE transactions on pattern analysis and machine intelligence*, *31*(7), 2009, 1195–1209.

Zhang, S., Zhu, Q., & Roy-Chowdhury, A. (2016). Adaptive Algorithm and Platform Selection for Visual Detection and Tracking. , 2016, 1–10.

Zhong, W., Lu, H., & Yang, M.-H. (2012). Robust object tracking via sparsity-based collaborative model. *Computer vision and pattern recognition (CVPR), 2012 IEEE Conference on* (pp. 1838–1845), IEEE, 2012.

Zhu, G., Porikli, F., & Li, H. (2015). Tracking randomly moving objects on edge box proposals. *arXiv preprint arXiv:1507.08085*, , 2015.

Zitnick, C. L., & Dollár, P. (2014). Edge boxes: Locating object proposals from edges. *European Conference on Computer Vision* (pp. 391–405), Springer, 2014.