Electronic Monitoring: Best Practices for Automation

BENJAMIN WOODWARD
MARK HAGER
HEATHER CRONIN

Gulf of Maine Research Institute
Science. Education. Community.

Table of Contents

Introduction

This document is a reference for designing an Electronic Monitoring (EM) system suited for automated or semi-automated video analysis. It is not meant to be an exhaustive review of EM systems, and will focus only on aspects which can make or break automation capabilities. It presents a broad range of information for use as a guide to industry groups, governments, or EM vendors considering or currently developing automated EM programs. Finally, it provides technical detail for groups which have experience in automated EM programs and an interest in the lessons learned from New England.

Automation of visual imagery analysis has progressed at an amazing rate in the last decade. Many common tasks can now be fully automated, and automation is even used to enhance human

performance of difficult tasks. This document will point out common EM tasks and how machine learning has had success or failure automating those tasks. It will also examine the relative benefits of automating certain tasks and highlight EM system designs which enable task automation.

The below sections discuss EM system design for physical constraints that drive camera requirements and placement, hardware specifications for processing needs, and algorithm considerations for data collection goals, annotation workflows, and expected performance.

System Level Design

Summary:

The first step towards understanding the utility of automation in an EM program is to evaluate the program objectives and existing monitoring standards for the fishery. Because EM programs differ in data collection objectives, time and cost burdens and data standards, automation applications will vary across programs. The scope and level of detail captured by footage of tasks intended for automation will influence both EM system installation and algorithm performance. Cameras should be installed in locations which capture the best compromise between the scope of the activity, the necessary level of detail, environmental and structural factors, and camera specifications.

Program Design: Data Collection

Across EM programs, data collection varies according to a variety of factors. Program type, fishery, fishing method, vessel type, and targeted data categories may all affect how automation incorporates into an EM program. Further, the relative effort needed to collect targeted data categories influences how automation is applied and which tasks are designated for automation. Because of the high variance in automation needs between EM programs, we do not intend to describe all case-by-case specifics for program design. Rather, we provide generalizations from a broad range of experiences, with a few descriptive examples.

Considerations for Targeted Activities

Depending on program objectives, the EM system will target the capture of certain on-board or dockside activities. Data collection for all or a subset of those activities may be suited for automation. Common activities for which data collection is automated include, but are not limited to: setting of gear, hauling of gear, sorting of catch, discarding of catch, catch processing, catch stowing, measuring of fish, protected species interactions, and crew-specific behavior. These activities may occur in either discreet locations on the vessel or as mobile processes that analyses must track across the vessel or fishing area. Some may require detailed imagery for analysis and data collection, while others only require an overview image. Therefore, when considering automation during EM system design or installation, it is not only important to consider the targeted data categories, but also the manner in which targeted activities are conducted and the level of detail necessary to identify each activity. Installation of the EM system and areas monitored must be appropriate for the location and scope of the targeted activities and data collection categories.

Installation considerations

To implement machine learning across a wide range of activities and conditions, EM system installers should make deliberate decisions regarding the camera field of view, camera distance from the object or scene, angle of incidence, physical mounting, and environmental conditions surrounding the camera. The following paragraphs provide suggestions from our experience of best practices for each of these parameters when designing an EM system for automation.

Field of view

The camera field of view should be as narrow as possible to see all relevant information about the targeted activity. For an activity such as sorting or gear retrieval on board a trawl vessel, this may need to be relatively broad, while for gear retrieval on a gillnet vessel, the field of view may be more limited. Finally, if the targeted activity includes detection, counting, identification, or measurement of fish on a measuring board, a camera with a field of view limited to the only measuring board is ideal.

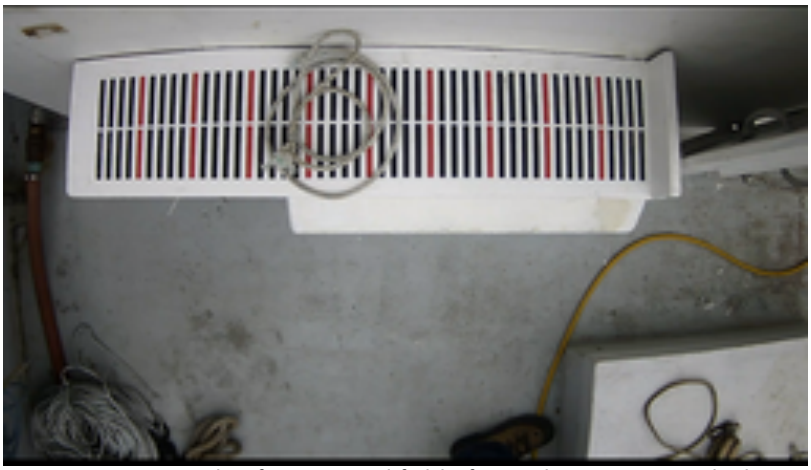

Figure 1: An example of a restricted field of view that captures only the measuring strip and its immediate surroundings. This tight view limits the amount of noise an algorithm needs to accommodate in the area surrounding the activity of interest. Image provided by the NE groundfish audit project.

Angle of Incidence

The best angle for most EM applications is as overhead of the targeted activity as possible. This allows for a consistent size and distance of all objects, regardless of their position within the camera field of view. On many vessels, this is achieved by mounting cameras in the rigging or A frame for broad deck shots. Close shots, such as species ID and length measurements, are best achieved by positioning discard control points or measuring strips near vessel structure such as wheelhouse overhangs. This creates both a restricted field of view and an overhead shot for detailed imagery analysis. In cases where discarding or measuring must occur away from any overhead vessel structure, fabricating a structure (e.g. a mounting arm) can create a view which achieves a top-down image.

Figure 2: An overhead camera mounting which generates a broad, top-down view of sorting activities. Image provided by the NE groundfish audit project.

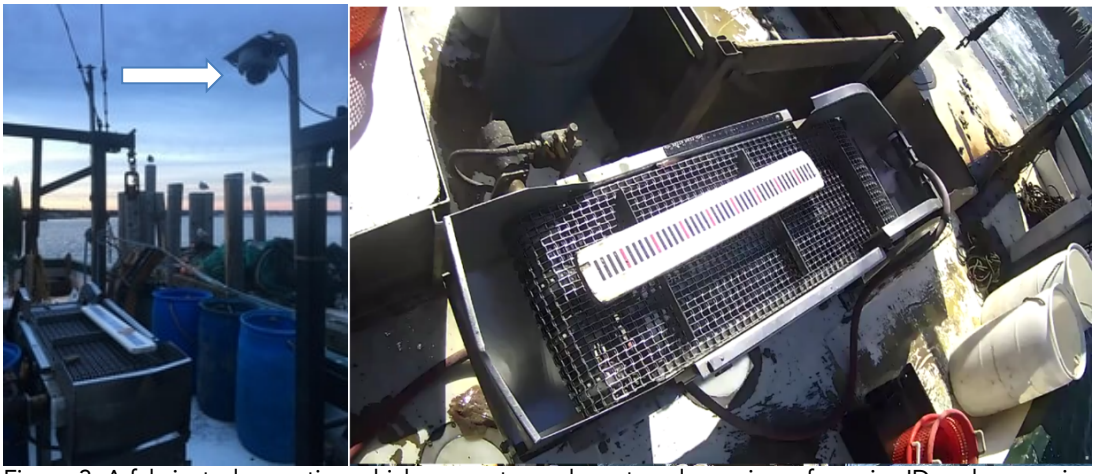

Figure 3: A fabricated mounting which generates a close, top down view of species ID and measuring. Image provided by the NE groundfish audit project.

Camera Distance to target

Once again, the target and objective of the camera footage must be taken into account when thinking about the mounting distance from the target. The combination of the distance from the recorded activity, the resolution of the camera sensor, and the field of view provided by the camera's optics dictate the level of detail provided for a specific object or activity of interest. Most modern object detection algorithms are sized to work on objects that are between 50 and 450 pixels on a side, if one was to draw a square box around the object.

Therefore, the best way to ensure adequate detail for automation is to create tables to calculate the expected object size based on the real size of the object, the distance between the object and the camera, and the camera sensor and lens parameters (Table 1). Calculations in Table 1 show workflows for determining system requirements using the example of a 1-foot length object at a range of 25 feet, that must be 100 pixels in size in the captured image. They solve for the required lens focal length for a given sensor type or the object size, in pixels, given a specific sensor/lens combination. The first is useful in determining the appropriate components for an EM camera setup, while the

second is useful to evaluate the appropriateness of a specific EM setup. A useful lens calculator for making FoV calculations can be found here: https://www.1stvision.com/lens/fov-lens-calculator. For descriptions of object and sensor parameters, see the "Camera Design and Configuration" section on page 8.

Object and Sensor Parameters	
Sensor Pixel Pitch	6.4 µm
Sensor Format	1/1.2"
Sensor Horizontal Size	10.67 mm
Sensor Vertical Size	8mm
Minimum Desired Pixels on Object	100
Object Distance from Sensor	25 ft
Object Length	1 ft
Camera + lens focal length	16mm

Computations		
Goal	**Calculation**	**Result**
Object size on sensor	$= Sensor\ pixel\ pitch * Minimum\ desired\ pixels$ = 6.4µm * 100	0.64mm
Required focal length	$= \left(\frac{object\ distance}{object\ length}\right) * Object\ size\ on\ sensor$ = (25 / 1) * 0.64	16mm
Resulting Horizontal FoV	$= 2 * \arctan\left(\frac{seonsor\ horizontal\ size}{2*Required\ focal\ length}\right)$ = 2 * atan(10.67 / (2 * 16))	36.9°
Resulting vertical FoV	$= 2 * \arctan\left(\frac{Sensor\ vertical\ size}{2*Required\ focal\ length}\right)$ = 2 * atan (8 / (2 * 16))	28.1°
Resulting diagonal FoV	$= 2 * \arctan\left(\frac{\sqrt{Sensor\ horizontal\ size^2+Sesnor\ veritcal\ size^2}}{2*Required\ focal\ length}\right)$ = 2 * atan(√ (10.67^2 + 8^2) / (2 * 16))	45.2°
Object size on image	$=\frac{Object\ sensor\ size}{Sensor\ pixel\ pitch}$ =0.64mm/ 0.64µm	100 pixels

*Note: Computations assume "pinhole" camera model, meaning lens distortion is ignored. Reasonable assumption for lenses with ≥12mm focal length.

Table 1: Calculations for determining expected object size, ideal distance between the object and the camera, and the camera sensor and lens parameters. Example calculations are for an object 1-foot in length.

For broad shots intended to perform activity recognition, object detection and/or object tracking, a camera mounted 10-30 feet from the target is usually suitable when using 1920 X 1080 resolution. For the same resolution we have found that automating the identification of species in the 10-70 cm range requires a closer view of 3-15 feet.

Physical Mounting

The point at which the camera connects to the vessel is critical. For all automation applications, the mounting location should experience minimal vibration. However, during normal vessel operation, cameras will experience acute vibration during fishing activity and chronic vibration during steaming. Cameras commonly experience acute vibration when mounted near winches or booms and chronic vibration when mounted to the vessel superstructure. Mounting to high vibration areas such as these is most easily avoided through discussion with the vessel operator. In some cases, it may even be prudent to perform mock gear hauling, setting or hoisting to mimic vibration during fishing activity. While certain cameras may be robust enough to handle both acute and chronic vibration, other models may need a form of vibration absorption. For these, we have found ¼ inch neoprene pads to reduce the effects of vibration. For success with automation both types of vibration are important to avoid or mitigate. Specifically, for objects identified obtaining a single frame associated with an annotation, it is increasingly difficult to obtain such a frame when vibration distorts the imagery.

Another consideration for camera mounting, is minimizing the potential for gear to collide with the camera. Cameras are easily struck and moved off target by gear used during fishing and offload. Therefore, the best places to mount cameras may be on the back of the wheelhouse, which protects cameras from moving gear. If it is essential to mount near moving gear, cameras are most protected by

mounting opposite from the moving gear on beams or rigging. A close look at wear and tear on the vessel can help to determine locations of frequent gear contact. Taking time to ensure the crew won't bump the cameras when mounting lower to the deck is also important. While it may be obvious that reduced physical interactions are useful for regular EM practices, such interactions also create incomplete or inconsistent automation training data sets because they change the position of or entirely obscure the target activity.

Environmental Conditions

It is important to gather and annotate training footage under the same environmental conditions as are expected in operational footage. This ensures good generalization performance by algorithms, especially for object detection, object tracking, and activity recognition. For example, deck lighting on the vessel used to create algorithm training data sets should be like that used by the rest of the fleet. It may seem advantageous to use training data from a vessel with the best lighting or to add additional lighting to the vessel to improve training data, but building algorithms under ideal conditions can ultimately degrade generalization performance. Therefore, when designing EM systems for the collection of training data, the characteristics and environmental conditions on the training vessel should be representative of the fleet.

Alternatively, algorithms for tasks such as classification of fish within a pre-determined bounding box (e.g. the green box in Figure 6), may be trained using data captured under unrepresentative conditions. For example, if an algorithm can draw bounding boxes around all fish, but not identify species, a second algorithm trained on close-cropped images of fish can identify the sub-images created by the bounding box algorithm. These algorithm pairings are much more robust to different environmental conditions (e.g. pictures of fish in a lab vs on deck).

The following sections expand on specific camera parameters, and how to evaluate them for your application.

Hardware Specifications

Summary

EM systems typically consist of an array of machine vision cameras. The camera configuration will determine the automation capacity of footage collected from each camera in the system. Desired camera sensitivity, image blur, and field of view are achieved by balancing the camera configurations to the optimal level for the captured scene or activity.

Camera Design and Configuration

Unlike cameras meant for consumer applications, cameras used in EM are from a class of cameras called machine vision cameras. Machine vision cameras lack the components meant for human interaction, such as a shutter release or viewfinder, and consist of four main components.

- **Lens**: a system of optical components that focus light onto the imaging sensor.

- **Imaging sensor**: an array of sensors that convert light into electrical signals.
- **Readout electronics**: turn an analog electrical signal into a digital signal, and may apply various transformations such as amplification, filtering, or binning.
- **Interface**: transfers image data to a computer for recording or processing and accepts commands from a software application.

Key requirements that drive design of camera systems include the following:

- **Field of view**: the angular span corresponding to the image, specified as the diagonal, horizontal, and/or vertical field of view.
- **Blur tolerance**: the acceptable amount of blur for objects of interest. For moving objects this is dominated by motion blur.
- **Object size**: the object of interest's dimensions in an angular sense.
- **Frame rate**: the frequency at which the camera acquires consecutive images.
- **Lighting conditions**: the required lighting conditions for proper camera function

The camera's sensor format and lens determine the field of view. The sensor format refers to the physical dimensions of the imaging sensor, which determine aspect ratio. Many machine vision cameras have interchangeable lenses which use C-mount or CS-mount as the interface between the camera case and the lens. For hardware redundancy, it is possible to use the same image sensor for all cameras but achieve different fields of view by using different lenses in each camera. Other cameras are designed with the lens and image sensor packaged together. These often have an optical zoom capability to change the field of view between cameras or dynamically during monitoring activities. However, per the above notes on vibration, optical zoom motors must be sufficiently ruggedized for the operating environment. See Table 1 for an example of how field of view affects automation capabilities.

Image blur is an especially important consideration when designing an EM system for automation. Just as a blurry image can obscure important details for identification by a human, the same is true for an algorithm. Therefore, installations should aim to balance the above parameters against tolerance for expected motion within an image.

Depending on automation goals, tolerance of blur will differ. For simple activity or motion-based actions, a higher level of blur is tolerable because algorithms will use scene context to make decisions. For object detection and classification, it is important to drive motion blur close to zero because algorithms will use fine-grained details such as morphological structure or coloring pattern to make decisions. While simply reducing the amount of time that the camera aperture is open will reduce image blur, it can also create images that are overly dark, and do not have the detail required to make accurate classification decisions. For many human video review scenarios, these challenges can be overcome by tracking the fish through the whole scene and piecing together identifying information from different parts of the scene. While this is technically possible for an algorithm, it requires a lot more training data and a more complicated architecture. Therefore, the recommendation is to have as high quality an image as possible for every frame of interest.

The goal of reducing image blur is in direct tension with higher frame rates (high detail) and low light scenarios. The reason for this is that to overcome low light and achieve higher detail, the camera aperture is left open for as long as possible. Many consumers observe this effect in low light cell phone pictures. The following lists the main parameters that increase the ability of a camera system to take high quality images with minimal motion blur and highlights recommendations for automation.

- **Pixel pitch**: the physical size of each pixel on the sensor.
    - *Recommended: larger is better.*
- **Quantum efficiency**: the percentage of incident photons converted to an electrical signal.
    - *Recommended: higher is better.*
- **Aperture**: the size of the smallest window through which light passes in the lens.
    - *Recommended: see f-number.*
- **F-number**: Aperture is often expressed as a ratio of focal length to diameter of the entrance pupil called f-number. F-number is favored over simply using aperture diameter/area for characterizing lens "brightness" because image illuminance is directly proportional to the square of the f-number.
    - *Recommended: as a rule of thumb, the lower the f-number, the better the lens for low light or high motion scenarios.*
- **Exposure**: the length of time the sensor accepts light for each image.
    - *Recommended: Longer exposure risks more motion blur. Use the lowest possible exposure time to get usable imagery.*
- **Lens transmissivity**: the percentage of incoming light that reaches the image sensor.
    - *Recommended: Higher is better.*

Algorithm Considerations

Summary

Algorithms used for EM generally fall into one of two categories: object identification or activity recognition. Object identification is used to recognize an object within a scene and possibly perform some task related to that object. Activity recognition is used to identify instances of a particular action or activity occurring in the video footage. Activity recognition and object identification algorithms are able to build off of each other to perform more complex tasks and further reduce the burden to human review, but as the level of automation increases, so does the level of difficulty in creating that automation pathway. Below we describe automation pathways and how they might work together for a number of common EM tasks including: object identification, fish presence recognition, fish species identification, fish counting, feature measurement, and various activity recognition scenarios.

Data collection for these pathways requires careful EM system design and installation as well as considerations for building training data sets in human-review workflows. For spatial tasks, video reviewers may draw a shape or point on the scene, while for temporal tasks, reviewers may mark a series of timestamps.

Possible Algorithm Tasks

Broadly, algorithms for EM fit into two overlapping categories: object identification and activity recognition. Algorithms will fall into one or both categories depending on purpose and use, and it is important to identify automation objectives when choosing either category. In object identification, the goal is to recognize the presence of an object within a scene (e.g. fish species of interest), and then localize, identify, or measure that object. An example of this is the Northeast Multispecies Audit model EM program. In this model video reviewers count, identify the species, and obtain a length measurement of each fish in imagery from cameras placed above measuring boards.

Figure 4: A fisherman measures discarded fish, using a measuring strip installed below the EM camera. The fish length, and calculated weight is captured during video review using this image. Image provided by the NE groundfish audit project.

The second algorithm category is activity recognition (AR). In this case, the goal is to identify when a specific activity is happening within a video scene (e.g discarding of fish, loading or unloading of the hold, etc.). An example of this is the Northeast Multispecies Maximized Retention model EM program. In this model the goal is to identify when fish are brought on board, when they are unloaded, and to identify any discarding of allocated groundfish. Figure 5 shows the outputs of an AR algorithm that has characterized the different activities of a fishing trip. The red bars denote start and stop times of activities as recorded by a human reviewer. The shaded bars denote recognition of specific activities by algorithm. Outputs such as this have the potential to both focus human video review efforts and reduce the amount of stored video footage by differentiating between high-priority and low-priority video. The algorithm is able to have a granular description of the different activities (e.g. Catch Sorting is broken into Sort and Hold Loading), so that even with less than optimal performance, someone reviewing this graph can easily pinpoint where to look for certain activities.

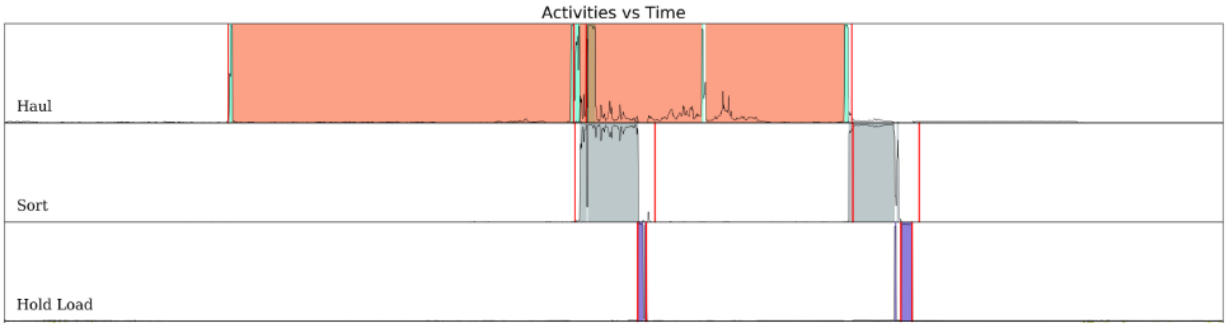

Figure 5: An example of output from activity recognition algorithms. The red bars denote start and stop times of activities as recorded by a human reviewer, while the shaded areas denote activity duration as identified by AR algorithms. Algorithm outputs: Orange segments represent calculated haul time. Green segments denote net handling activity and occur on either side of the haul as the net is set in or hauled out. Gray segments denote when catch handling is occurring. Purple segments denote video in which catch is being loaded into the hold.

Object identification and activity recognition algorithms have even greater potential when used in tandem to perform a specific task. Figure 6 shows how different automation tasks might build off of each other to reduce review time. The difficulty associated with implementing hierarchical algorithms increases as the level of automation and task-specificity increases. Algorithm implementation must therefore balance the potential reduction in human-review effort and algorithm specificity.

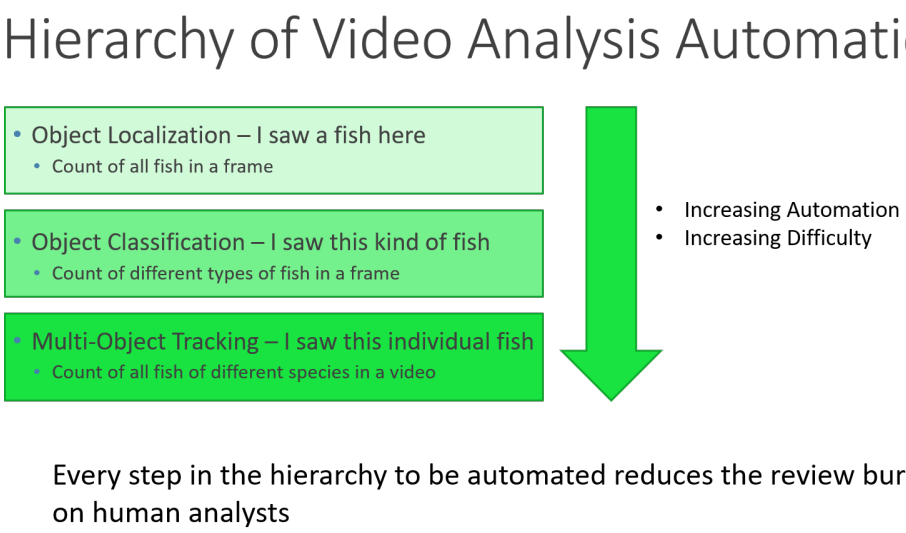

Figure 6: Hierarchy of video analysis. As the automated task becomes more specific, automation difficulty and potential reduction in human-review effort increase

Object Identification

The task of object identification within a scene is usually associated with image recognition tasks. For this reason, most object identification workflows are applied frame by frame, with additional algorithms layered on top (e.g. tracking algorithms for counting fish). For these workflows it is important to have a training data set that has annotations associated with individual frames. Most modern object detection algorithms use Convolutional Neural Networks (CNNs) as their baseline. Common implementations for video are YOLO and Faster R-CNN, though there are many others, each with their own balance of accuracy, inference speed and processing requirements.

The following paragraphs describe examples of algorithm workflows, proceeding from least complicated to most complicated. These are intended to provide examples for designing workflows to meet an EM program's needs. There is almost nothing unique about the example of identifying fish in the following, and all of examples provided would be equally valid for just about any EM application, regardless of species (even humans, which is where most of the technology comes from).

Fish Presence Recognition

One of the simplest workflows is to recognize points in a video in which fish are present within the frame. If an algorithm reliably recognizes those points, then a workflow that requires human analysts to work with video containing fish may be drastically reduced by allowing a reviewer to jump from event to event rather than combing through video to locate fish-handling activities. The algorithm for this case does not need to make judgments such as the species of the fish or measure any physical property such as length. The training requirements are much more lenient for such a workflow, and the performance and reliability of the resulting models are generally very high.

With a slight modification this workflow can accomplish fish presence recognition within a Region of Interest (ROI) of an image. Usually the ROI is some subset of the image such as a measuring board or conveyor belt. In some cases, fish presences recognition can even be accomplished without the aid of an additional algorithm. There are many software tools available to interact with images in this way (e.g. OpenCV, Python Imaging Library), and integrating them into review software is not a difficult task. Figure 7 shows an example of a fish recognition (green box) performed inside a defined ROI (blue box). The ROI reduces both processing load, as well as the potential for false alarms. In this example, the algorithm is implemented as part of the open source OpenEM algorithm and software library.

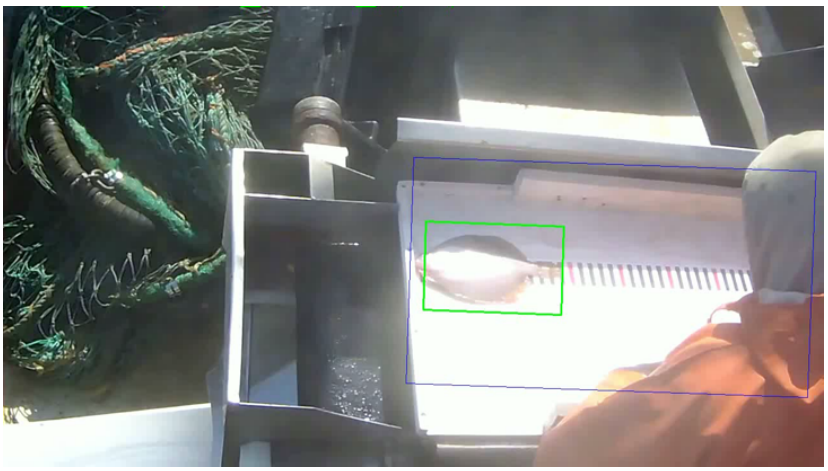

Figure 7: Image with a defined Region of Interest (ROI) in blue, as well as a localized fish in green. By restricting an algorithm's field of view to a specific ROI, performance is improved by excluding potential confusing scenery (e.g. background, totes with other fish in them, etc.). Image provided by the NE groundfish audit project.

Fish Species Identification

A step up from the previous workflow would be to identify the species of a fish within a frame. This usually requires some concept of localization within a scene because in many use cases there are

multiple fish of multiple species types within a scene. Performing localization requires the use of more complex algorithms which need increased amounts of training data that is harder to generate.

To build training data sets, human reviewers localize objects within a scene by describing the bounding coordinates for the object. Typically, coordinates are represented as a box, but any polygonal shape or even a pixel mask can be used. This information is used to train a localization algorithm to judge itself when it creates bounding box proposals. By giving it examples of localizations (boxes), it learns to draw the appropriate boxes in new images it is not trained on. For imagery, there are many free tools available to generate these types of localizations. A few of them are Tator (https://github.com/cvisionai/Tator), CVAT (https://github.com/opencv/cvat), LabelMe (https://github.com/wkentaro/labelme), and many others. In the above example, (Figure 7) the green box shows an example of a fish localized within an image.

Fish Counting

Fish counting is an example of a task in which information needs to be propagated from frame to frame to determine something about the video. Typically, tracking is incorporated to recognize the same fish from frame to frame and when it enters and exits the footage. Tracking requires a concept of localization as in the fish species recognition but it does not require a full species identification algorithm. For cases with one or a few fish with little to no overlapping or long-term occlusion, there are many tracking algorithm options. Cases with more than a handful of fish and high potential for occlusion (e.g. on a conveyor belt) are significantly more difficult. This is known as the multi-object tracking problem, and it is not a well solved problem in most domains. The training data required for multi-object tracking is vastly harder to generate by hand. Most efforts in generating multi-object tracking algorithms are a combination of automation assistance to generate training data as well as model-based tracking algorithms to reduce the amount of required data. With the resources and the patience though, the benefits of generating high quality tracking training data are immense.

Feature measurement

Oftentimes there is a desire to not just count and identify, but to also generate a feature measurement (e.g. length) for objects of interest. Depending on the feature and the scene, this task has varying degrees of difficulty. For example, fish that are on a fixed measuring board with a fixed-distance camera view are straightforward to measure accurately. In this instance the algorithm only needs to measure the pixel distance, which is usually well approximated by the bounding box generated by the object detection algorithm around the fish (see Figure 7). That pixel distance is then converted to length units using a fixed conversion factor.

It is significantly more difficult to generate a feature measurement in scenes that do not have a fixed distance or canonical view of the fish (e.g. when fish are hauled over the side of a boat). In this case it is usually required to have a stereo camera set up, which needs much more complex algorithms and training data. An example of work being done for stereo measurement is in the Alaska Fisheries Science Center, in conjunction with the University of Washington (https://www.fisheries.noaa.gov/feature-story/advancing-innovative-technologies-modernize-fishery-monitoring).

Activity Recognition

Activity recognition in video is most often associated with labeling of short video clips (YouTube-8M, Thumos, etc.), but for use in longer, unconstrained video clips, it is usually known as event detection. This is still a very active area of research, and there are not common frameworks to fall back on for this task.

The good news for EM is that the required bounds for event recognition are often much looser than in other domains. For instance, if an algorithm can narrow down to +/- 1 minute for activities of interest, this is usually sufficient for EM purposes. This is especially true if a human will be using those time stamps to jump to parts of video and manually verify events of interest. This level of temporal accuracy is also likely acceptable for use in CPUE or other fishery metric calculations.

Typically, activity recognition algorithms also have less burdensome annotation requirements for building training data sets because they only requiring start and stop timestamps for events of interest, instead of annotations such as bounding boxes, lines, or other shapes. These are often already recorded in review software and require little modification of review technique to generate appropriate training data for algorithms.

The following paragraphs describe a few common workflows for activity recognition.

Fishing Activity

In this use case the goal is to identify portions of video that contain fishing activity of interest. The intended output would be a timestamp, duration, and label. For most activities of interest (e.g. setting or hauling of gear or catch handling) the accuracy can be close to 100%, with relatively tight time bounds. In the case of a 24/7 camera system, this can eliminate close to 99% of the captured video as not relevant, resulting in significant storage and transmission savings, as well as review time savings. When targeting this level of automation, it is best to have clear camera views of all areas where targeted activities can occur. The best views tend to be overhead, with resolution on the order of 720p or better, given the typically available mounting points are higher up on the boat. This is a rule of thumb and depending on available mount points and required fields of view, can be higher or lower (see "Hardware Specifications", page 7).

Hold Loading/Unloading

For non-fishing activity such as hold loading or unloading, activity recognition starts to cross over into object detection or have otherwise more difficult aspects. For instance, EM protocols may aim to determine when a hold is loaded, count the number of totes loaded, and identify the species associated with each tote. This is a compound activity that comprises the recognition of the loading event, and then counting of totes and species identification. The loading event can be bounded with simple time stamps but counting of totes and species identification additionally require object detection and tracking.

Typically, this type of activity recognition needs an unobstructed camera view of the area that contains all objects of interest and their intended destination. Again, this is best achieved through an overhead camera view. For adequate views of individual totes and individual fish within totes, a

higher resolution such as 1080p is often required, due to available overhead mounting positions. This is due to the algorithmic specification of an object subtending at least 50 pixels of the field of view.

The data annotations needed to achieve all three of these goals are a little more involved than the time stamp only requirements for fishing activity and are similar in effort to object detection annotation.

Anomalous Activity

Oftentimes it is desired to flag anomalous events such as a protected species interaction, prohibited discarding, crew in a dangerous position, or other activity that is of special interest to managers or captains. The diversity of possible events in this regime means that detecting these events is usually a noisy problem, with a fair number of false positives. It is possible to construct high performance algorithms with low false positive rates and high recall, but they tend to require more training data than is typically available and cost more to develop than is justified by what they are detecting.

With the above considerations in mind, a useful paradigm for anomalous activity identification is to identify a region of interest (ROI) where these activities are likely to occur and to focus processing on that area. The requirements are then an unobstructed view of the area, with an overhead view typically desired. In the case of identifying activities such as operations in a dangerous or restricted area, lower resolutions such as 720p can usually be accommodated. However, for the use case of identifying illegal discards or protected species interactions, there is an element of object detection and identification that necessitates a higher resolution. Discriminating between allowable and prohibited discards is usually an easier proposition for an algorithm than full species identification. Protected species interactions are easier still but usually suffer from a lack of training data.

Data Collection

For all the use cases described in the previous section, there are some important differences between systems designed for human-only video review instead of algorithm-assisted video review. This section will highlight some of the footage requirements from automated use cases, as well as give examples of considerations that may not seem obvious when designing for human analysis rather than algorithm analysis. In addition to requirements on the EM system for operational functionality, there are also requirements on the data collected to train algorithms to ensure they are robust and perform as expected in application. The following is a list of guidelines for getting the best EM footage for automation

Camera Frame Rate

Generally, human reviewers are adept at inferring information between frames of video and can make more robust judgements than algorithms regarding an object of interest by using multiple partial views. This allows human reviewers to accommodate slightly lower frame rates than automated analysis.

Camera frame rate is primarily important when the objects of interest are only in the field of view for a short amount of time. For quick moving operations such as a conveyor or a discard measuring operation, we therefore recommend 15 frames per second as a minimum frame rate to guarantee a

good view of the object. For activity recognition, where judgments are made closer to the minute time scale, it is often enough to capture 5 fps, or sometimes even 1 fps footage.

Object placement and occlusion

Humans are also particularly adept at tracking objects through occlusions (e.g. hand covering) and using context from separate views to piece together classification clues. For instance, a human reviewer may be able to easily identify a fish that is mostly occluded by a hand when being measured by looking back a few seconds in time to find a better view. Algorithms do best when they are afforded a single, clear view at a known point in space or time. This minimizes the need for complex tracking algorithms, or algorithms that consider every frame in which they see a particular fish. While these types of algorithms are sometimes necessary, a significant amount of development time, and time spent generating training data can be saved if you can provide that "money shot" view. Occlusions are best avoided through training of vessel crew and a thorough discussion of deck operations during system installation.

Data Annotation

As has been discussed in previous sections, the primary method of training algorithms is a technique known as supervised learning. This is accomplished by showing the algorithm many positive and negative examples for the concept that it is learning. Algorithms will use various types of reviewer-generated annotations as the cues to understand what is meant by these examples. For spatial tasks, such as species detection and recognition, annotations may be drawn on the scene. For temporal tasks, the most common annotation is either a single time stamp along with a label, or a start and stop time, also with a label. The following is a list of spatial annotation types, and where to use them.

Box

This is the most common annotation to use in visual analysis tasks. Box annotations describe the coordinates in a scene that create a rectangle that encompasses the object of interest. For instance, a rectangle may be described by the x and y coordinates of its four corners, or by the x and y coordinates of one of the corners and a width and height. It does not matter which convention is used so long as the same convention is always used, and the algorithm is provided a description of the convention.

Line

Line annotations are most often used to measure an object's length. Usually, they are used in conjunction with box annotations. The best way to describe a line is the x and y coordinates of its endpoints.

Dot

The dot annotation most commonly used to denote an area of interest or to count objects within a scene. It is usually used with more complex algorithms that attempt to infer something about the location marked by the dot by looking at the whole scene.

Pixel mask

Pixel masks are a tool used to generate segmentation masks, which seek to identify all pixels belonging to an object in a scene. With a complete representation of an object, algorithms can try to learn things such as morphometrics or pose of an object. Pixel masks are typically represented as a list of pixel indices within an image, where a defined convention determines how to assign numbers to pixels within the image.

Polygon

Polygons may be used in place of pixel masks, as a lower fidelity form, to accomplish the same types of tasks. It is far easier to draw a polygon than it is to color in every pixel belonging to an object. Polygons are usually represented as ordered lists of the x and y coordinates of vertices.

On Vessel Implementations

Automation algorithms may not only have different goals for video analysis but may also have different modes of operation. One extremely useful use case is running algorithms on the vessel on which the video footage is captured. This can drastically reduce both the storage and transmission requirements for EM video data. However, the available compute capability on board a vessel is usually significantly less than what is available at a data review center, and so there are considerations for expected performance and reasonable use cases.

The main considerations for on board vessel algorithm evaluation revolve around Size, Weight, and Power - and Cost (SWAP-C). It is technically possible to put a powerful enough computer on board a vessel to run the types of algorithms discussed in this document, but those computers can range in cost up to several thousands of dollars. Additionally, without optimization, many of the algorithms developed do not run in (near) real time.

There does exist a growing field of low cost, lightweight, ruggedized computers for machine learning inferencing (e.g. NVIDIA's Jetson platform). These systems are sufficient for running the following types of workloads:

- Object detection and classification
- Activity Recognition

These two workflows comprise a large majority of the types of automation that are most useful aboard a vessel, because they can act as trigger points for either prioritizing recordings, or human intervention, via lightweight message outputs. It is much simpler to consider sending a small text file with a summary of the last minute's worth of activity or discards over satellite or some other cellular link than it is to send an entire minute's worth of multi-camera video frames.

Conclusion

In Electronic Monitoring programs, automation is increasingly used to create efficiencies for common EM tasks. Data collection for these tasks requires careful EM system design and installation as well as

considerations for building training data sets in human-review workflows. Because EM programs differ in data-collection objectives, time and cost burdens and data standards, automation applications will vary across programs. Automation algorithms may not only have different goals for video analysis but may also have different modes of operation.

Whether looking to incorporate automation from the outset or building a program that may use automation down the line, considerations for machine learning should be incorporated into program and EM-system design on the front end. These considerations include camera configuration, installation limitations, scope and level of detail captured by footage, processing needs, data collection goals, annotation workflows, and the required level of performance. Cameras should be installed in locations which capture the best compromise between the scope of the activity, the necessary level of detail, environmental and structural factors, and camera specifications.

Algorithms used for EM generally fall into one of two categories: object identification or activity recognition. Object identification is used to recognize an object within a scene and perform some task related to that object. Activity recognition is used to identify instances of a particular action or activity occurring in the video footage. Activity recognition and object identification algorithms can build off of each other to perform more complex tasks and further reduce the burden to human review, but as the level of automation increases, so does the level of difficulty in creating that automation pathway. While object identification and activity recognition algorithms have even greater potential when used in tandem, algorithm implementation must balance the potential future reduction in cost or time with the human-review effort required to generate algorithm specificity.

The primary method of training algorithms is a technique known as supervised learning in which the algorithm is trained by showing many positive and negative examples for the concept that it is learning. Therefore, there are some important differences between systems designed for human-only video review instead of algorithm-assisted video review. Human-review workflows should be created in a way that easily generate the appropriate training data for algorithms and provide enough training data to ensure algorithms are robust and perform as expected in application. Operating these AI training workflows may require more budget and should be planned for on the outset of a project if automation is a known goal.

In conclusion, automation has the potential to drastically improve EM program efficiencies and therefore their effectiveness. Careful thought and proper planning are essential to making the most of the powerful technology available. This document is a compilation of best practices and lessons learned from the authors experiences implementing automation in their respective EM programs. We hope this can serve as a guide to help current and future EM practitioners make decisions that allow for successful automation in their program.