1.7

Received: 8 February 2022 | Revised: 13 August 2022 | Accepted: 29 August 2022

DOI: 10.1049/ipr2.12636

The Institution of Engineering and Technology WILEY

ORIGINAL RESEARCH

Motion stereo at sea: Dense 3D reconstruction from image sequences monitoring conveyor systems on board fishing vessels

Mark Fisher[1] | **Geoffrey French**[1] | **Artjoms Gorpincenko**[1] | **Helen Holah**[2] | **Lauren Clayton**[2] | **Rebecca Skirrow**[3] | **Michal Mackiewicz**[1]

[1]School of Computing Sciences, University of East Anglia, Norwich Research Park, Norwich, UK (Email: g.french@uea.ac.uk; A.Gorpincenko@uea.ac.uk; M.Mackiewicz@uea.ac.uk)

[2]Marine Laboratory, Marine Scotland Science, Aberdeen, UK (Email: Helen.Holah@gov.scot; Lauren.Clayton@gov.scot)

[3]Centre for Environment, Fisheries, and Aquaculture Science, Lowestoft, Suffolk, UK (Email: rebecca.skirrow@cefas.co.uk)

Correspondence
Mark Fisher, School of Computing Sciences, University of East Anglia, Norwich Research Park, Norwich NR4 7TJ, UK.
Email: mark.fisher@uea.ac.uk

Funding information
Horizon 2020 Programme, Grant/Award Number: 773521

Abstract

A system that reconstructs 3D models from a single camera monitoring fish transported on a conveyor system is investigated. Models are subsequently used for training a species classifier and for improving estimates of discarded biomass. It is demonstrated that a monocular camera, combined with a conveyor's linear motion produces a constrained form of multiview structure from motion, that allows the 3D scene to be reconstructed using a conventional stereo pipeline analogous to that of a binocular camera. Although motion stereo was proposed several decades ago, the present work is the first to compare the accuracy and precision of monocular and binocular stereo cameras monitoring conveyors and operationally deploy a system. The system exploits Convolutional Neural Networks (CNNs) for foreground segmentation and stereo matching. Results from a laboratory model show that when the camera is mounted 750 mm above the conveyor, a median accuracy of <5 mm can be achieved with an equivalent baseline of 62 mm. The precision is largely limited by error in determining the *equivalent* baseline (i.e. distance travelled by the conveyor belt). When ArUco markers are placed on the belt, the inter quartile range (IQR) of error in z (depth) near the optical centre was found to be ±4 mm.

1 | INTRODUCTION

To prevent over-fishing and wasteful discarding, the fishing industry is subject to regulation regarding catch composition. For example, the European Union's landing obligation regulations limit the length and species of fish that may be discarded. Human observers have traditionally been employed on board to verify compliance with these obligations, but recently Remote Electronic Monitoring (REM), also known as Electronic Monitoring *System* (EM), of fishing vessels has been used to augment and in some cases replace, observers in many fisheries worldwide [1].

Between 2008-2016, CCTV cameras were installed and used in trials of REM systems on board a number of Scottish fishing vessels and during this time there were attempts to obtain catch information such as species composition, numbers, lengths and volumes from the video footage [2]. However, manually counting, measuring and identifying fish species is laborious and time consuming, and this has motivated the development of a computer vision system, named Catch Monitor, designed to analyse the footage automatically [3, 4].

A key component of Catch Monitor is a convolutional neural network (CNN) capable of automatically identifying common species of fish landed by UK trawlers in 2D imagery from REM systems. Catch Monitor's success is founded on a large database of expertly labelled fish imagery, acquired by REM cameras overlooking conveyor belts on board UK trawlers and research vessels. The current database comprises more than 30 species, however while there are thousands of examples of common commercial species (e.g. Cod, Haddock) other species are underrepresented. We propose to address this by augmenting the training set using 3D models to generate additional novel 2D views. This motivates the investigation of motion stereo presented here. Additionally, 3D models offer possibilities for improved estimates of biomass (presently, weight and girth of fish are estimated by length).

Conveyor belt transport systems are common on board many trawlers, and are used to move fish to a workspace where they are gutted, sorted, packed and any remaining bycatch and fish guts ultimately discarded. The accuracy of automated counts and sizes of fish on conveyor belts depends on the density of the load due to the increasing number of occlusions. Successive CCTV video frames of fish in motion deliver multiple views of individual fish and with this brings the possibility of using Motion Stereo to render depth maps and 3D point clouds from a single REM CCTV video camera. Although work on Motion Stereo has been undertaken previously [5, 6] we believe our system is the first to be deployed outside the laboratory, on a commercial conveyor belt system that moves at variable speed, and with Complementary Metal Oxide Semiconductor (CMOS) cameras using rolling shutters.

Automated inspection and monitoring of material on conveyor belts is common in many industries [7, 8] and increasingly important for waste recycling [9, 10] and agrifood [11, 12]. Our work demonstrates that, in these scenarios, RGBD imagery can be acquired from existing monocular cameras. Many thousands of hours of REM CCTV footage have been archived since REM was initially deployed on board Scottish trawlers, and while future REM may be upgraded to support 3D cameras the possibility of augmenting this with legacy CCTV footage is very attractive.

Motion Stereo comes with the advantage that since the stereo baseline depends on the distance the conveyor belt moves between acquisition of the left and right frame it can be adapted depending on the desired accuracy of the depth reconstruction. Another advantage, over more usual Structure from Motion (SfM) of 3D reconstructions from images collected by a single moving camera, is that the camera can be pre-calibrated, and images rectified by applying a precomputed homography. We use Zbontar et al.'s Matching Cost - Convolutional Neural Network (MC-CNN) to determine stereo correspondences [13] which is implemented on a GPU to achieve a throughput of approx. 5 fps. We evaluated Motion Stereo by building a laboratory model before testing on REM CCTV video acquired during sea trials on board Marine Scotland's research vessel MRV Scotia. The precision of the system depends on precise measurement of the stereo baseline (i.e. distance moved by the conveyor belt). ArUco markers, fixed to the belt of the laboratory model simplifies this task. For trials footage we estimate the baseline by computing sparse optical flow and deploy a Kalman Filter to track the conveyor belt motion.

Our work makes three novel contributions. Firstly, it represents the first and only attempt to develop a practical inspection system using Motion Stereo to reconstruct a 3D scene. Secondly, we provide the first quantitative evaluation of motion stereo. We measure the accuracy and precision of the motion stereo 3D reconstructions and compare them with those of a conventional binocular stereo system. Finally, our experimental results using MC-CNN trained on the Middlebury benchmark confirm the network's ability to generalise and achieve good results in a completely different domain.

Although, the performance of motion stereo in the laboratory approaches that of binocular stereo, the deployment of motion stereo in an operational context faces some additional challenges. Firstly, since motion stereo imagery is not synchronised (i.e. the left and right images are acquired at different times), the stereo baseline is determined by the distance moved by the conveyor belt between the acquisition of successive image frames. Commercial REM systems tend to reduce the frame rate of the stored video stream and this makes recovering the motion of the conveyor belt between successive image frames challenging. Secondly, since REM video cameras employ CMOS sensors and rolling shutters, further processing is needed to correct motion artifacts. Thirdly, much the UK fishing industry relies on smaller vessels and there is limited scope for changing working practice or space to accommodate a dedicated scanner (such as [14]). As such REM is deployed in an uncontrolled and largely unstructured working environment and the real-world nature of the footage presents significant challenges to the vision system.

The paper proceeds as follows: Section 2 reviews related work; Section 3 describes our approach and the pre-processing strategies we adopted for the Laboratory Reference System and an Operational System used to process MRV Scotia's REM CCTV video. Section 4 presents results and we draw the paper to a close in Sections 5 and 6 with a discussion and conclusions.

2 | RELATED WORK

2.1 | Computer vision within the fishing industry

Over the last three decades the fishing and aquaculture industry has pursued research to assist in identifying species and gathering counts and sizes of fish [15, 16]. Catch Monitor [3] is one of a number of technologies funded by the European Union's (EU's) SMARTFISH H2020 project [17] aimed at developing systems underwater and topside, to count, measure and classify fish. The approaches adopted topside tend to be fishery dependent. For example, Catch Monitor targets medium sized vessels and exploits deep learning to count and classify fish captured in 'real-world' 2D colour imagery from REM CCTV. In contrast, Catch Scanner [14] represents a dedicated solution using cameras and laser scanners within an environment where lighting and occlusions are managed. Examples of similar systems include AS3ID, designed to recover a 3D model of crustaceans as they travel through a light box on a conveyor belt [18], computer vision to measure tuna fish in REM video as they are hauled on board a longline trawler [19] and algorithms to separate individuals in images of fish packed in boxes [20]. Recent work has leveraged research by the Artificial Intelligence (AI) community, CNNs are now commonplace and Mask R-CNN [21] is the dominant technique for separating individuals in cluttered scenes [3, 19, 22, 23].

Stereo vision is key for non-invasive subsea measurement of fish length with invariance to their changing body shape. Garcia et al. [22] acquire stereo pairs using the Deep Vision system directly placed in the trawl [24], and propose a novel fish sizing system.

2.2 | 3D reconstruction from a single camera

The terms multiview Structure from Motion (SfM) and visual Simultaneous Location and Mapping (vSLAM) are used by the computer vision and robotics communities to describe techniques for reconstructing 3D models from images acquired by a single (roving) camera [25, 26]. These approaches are the focus of a large body of work addressing a fundamental problem in computer vision, initially formalised by Longuet-Higgins [27]. Before this, the term *Motion Stereo* was coined by Ramakant Nevatia to describe the extraction of depth information from a sequence of progressive views naturally available in applications such as industrial parts placed on a slowly moving conveyor belt [5]. With hindsight, Motion Stereo can be summarised as a type of Structure from Motion (SfM) where the camera is stationary and object motion is constrained.

Nevatia presented results of experiments conducted as part of the Stanford Artificial Intelligence Hand/Eye project from a laboratory rig comprising a single stationary camera observing objects placed on a rotating turntable. Two decades later Ens et al. revisited the problem and proposed multi-scale algorithms for real-time motion stereo, describing how these could potentially be applied in automated inspection using conveyor belts or assembly lines in a manufacturing environment [6]. Their approach considered the baseline as a series of n increments of distance b and developed an expression for disparity based on measured disparity increments and belt shifts. Results from a laboratory rig were presented and a parallel implementation was developed running on Transputers.

Stereo vision is a mature technology and numerous stereo reconstruction algorithms have been published, some implemented in hardware (e.g. [28]), and available in libraries (e.g. [29]). Trucco and Verri provide a good overview of early stereo algorithms and other 3D reconstruction techniques in their text book [30]. Scharstein and Szeliski [31] distinguish between four steps that most stereo methods perform, that is, matching cost computation, cost aggregation, disparity computation/optimization, and disparity refinement. Matching cost computation is very often based on the absolute, squared, or sampling the difference of pixel intensities or colors. However, since these costs are sensitive to radiometric differences, costs based on image gradients and Mutual Information have also been adopted. Many current algorithms are evaluated using the Middlebury and KITTI benchmarks [32, 33], and a comparison is given on the Middlebury [34] and KITTI [35] Stereo Pages. Algorithms that adopt a deep learning paradigm are consistently amongst the best performers. MC-CNN [13] represents an early attempt using a CNN to compute the matching cost for image patches. This is followed by a pipeline, first described by Hirschmuller [36]. Hirschmuller proposed Semiglobal Matching (SGM) for cost aggregation. This approach casts the problem of finding a globally optimal disparity image in an energy minimisation framework and uses dynamic programming to recover a semi-global optimal solution. Two disparity images are computed, by considering either the left or right image as reference. Comparing these allows occluded pixels to be identified and their disparity 'in-painted' by copying neighbouring values. MC-CNN achieved an average error of 3.82 in 2015, while the most recent recurrent CNN network submitted to CVPR'2022, achieves an average error of 1.09 [34]. Such architectures target state-of-the-art GPUs and can achieve performance in excess of 100 fps.

2.3 | Object tracking with rolling shutter cameras

Object tracking is another fundamental problem that has been investigated by many researchers. Tracking moving objects in surveillance video is a demanding application and the subject of several surveys, for example, [37–39]. Using a Kalman filter [40] to track SIFT features [41] is a robust strategy since SIFT is invariant to changes in scale, rotation and illumination. The following represent only a fraction of published work [42–44]. The majority of today's image sensors used in video cameras are of the CMOS type. In contrast to classical CCD sensors which employ global sensor readout, the image rows of CMOS sensors are read in rapid succession, and the sensor is reset electronically so there is no need for a mechanical shutter. This arrangement is called an electronic rolling shutter and leads to a rolling shutter camera model [45, 46]. Rolling shutter sensors present additional challenges for stereo matching and tracking algorithms and remains an active research area [47–51].

As a result of the motion constraint imposed by the conveyor system there is limited benefit in extending recent work in tracking, or rolling shutter compensation, and we are able to achieve good performance using a relatively simple rolling shutter model. The featureless surface of the white conveyor belt is a limiting factor for motion tracking, which we've addressed using engineered markers (it could also be addressed by placing sensors within the belt transport mechanism). Adopting a more highly ranked stereo matching network [34] could bring fewer mismatches and a performance gain. While our work does not extend any individual system components it does demonstrate an engineering solution, reports the accuracy that can be achieved, and compares this to that of a conventional stereo camera.

3 | MATERIALS AND METHODS

3.1 | Video dataset

The data set for this work was acquired in the laboratory and from commercial electronic monitoring systems fitted to UK trawlers and marine research vessels. The laboratory model was constructed comprising a conveyor transport system and low cost USB binocular camera [52]. Left and right cameras use a rolling shutter CMOS sensor and are not synchronised.

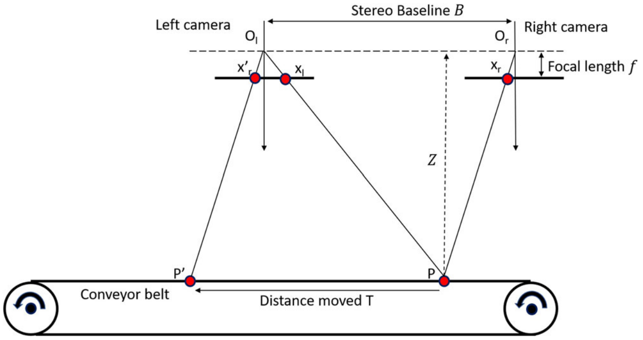


FIGURE 1 Stereo image formation

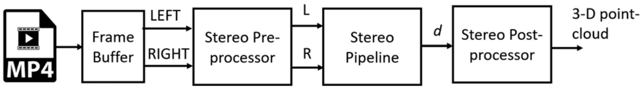


FIGURE 2 System architecture

The laboratory footage was captured in 1080p HD resolution and stored in MPEG-4 format. Operational video footage was obtained from REM systems provided by Archipelago Marine Research Ltd. [53] and Anchor Lab. [54]. These were installed on board Marine Scotland's research vessel MRV Scotia, and UK trawlers recruited to REM trials. The operational footage was captured in 800p HD resolution and stored in MPEG-4 format.

3.2 | Software

The software used in this work was written in Python and calls the following libraries: PyTorch [55], OpenCV [29], and Open3D [56]. The stereo pipeline front end uses a CNN proposed by Zbontar et al. [13] to compare 11x11 pixel patches. Subsequent pipeline stages comprise cross-based-cost-aggregation, Semi-Global-Matching (SGM), in-painting of mismatched/occluded pixels, and bilateral filtering. The pipeline is implemented for both CPU and GPU [57] hardware, however, when running on CPUs we usually omit cross-based-cost-aggregation as it is very time consuming.

3.3 | Stereo geometry

Consider a pair of pinhole cameras with parallel optical axis mounted above a conveyor system (Figure 1). Assuming the image planes are coplanar then by similar triangles [30]:

$$\frac{B}{Z} = \frac{B + x_r - x_l}{Z - f}. \tag{1}$$

Hence:

$$Z = \frac{Bf}{x_l - x_r} = \frac{Bf}{d}, \tag{2}$$

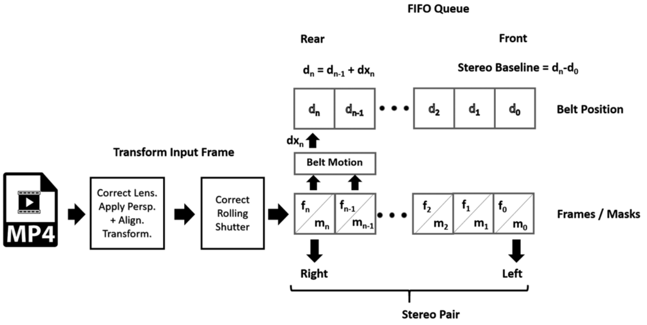


FIGURE 3 Frame buffer

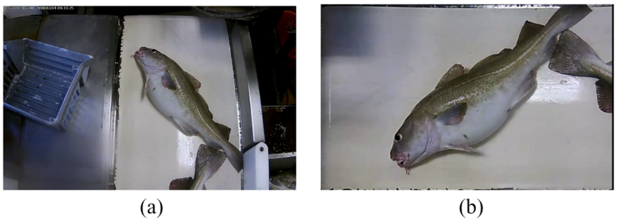


FIGURE 4 Video input (MRV Scotia): (a) Raw video frame; (b) Rectified image frame

where B represents the distance between the cameras (stereo baseline), f the focal length and d the disparity.

At a later time, when P is moved a distance $T = B$ by the conveyor belt, P' in the left image will be equivalent to that of P in the right. Again, by similar triangles:

$$\frac{T}{Z} = \frac{x_l - x'_r}{f}. \tag{3}$$

Hence:

$$Z = \frac{Tf}{x_l - x'_r} = \frac{Tf}{d}. \tag{4}$$

In general, the stereo cameras' optical axis will not be parallel, but from the fundamental matrix we can compute homographies that transform each image plane such that they are parallel [58]. Similarly, in general, the optical axis of a single camera will not be perpendicular to the conveyor belt but this can also be corrected by a suitable homography.

3.4 | Operational system

The main components of the system are shown in Figure 2. Two fundamental problems in computational stereo are correspondence and reconstruction [30]. The correspondence problem involves identifying parts of the left and right images that are projections of the same scene element. Using this information, together with knowledge of the geometry of the stereo system, reconstruction determines the 3D location and structure of the object. The matching algorithm implemented by the stereo

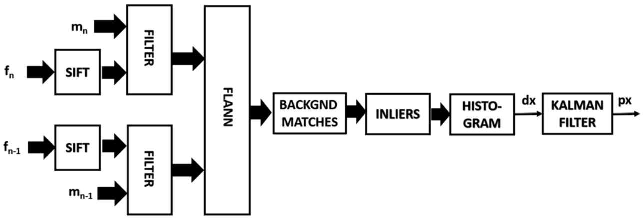
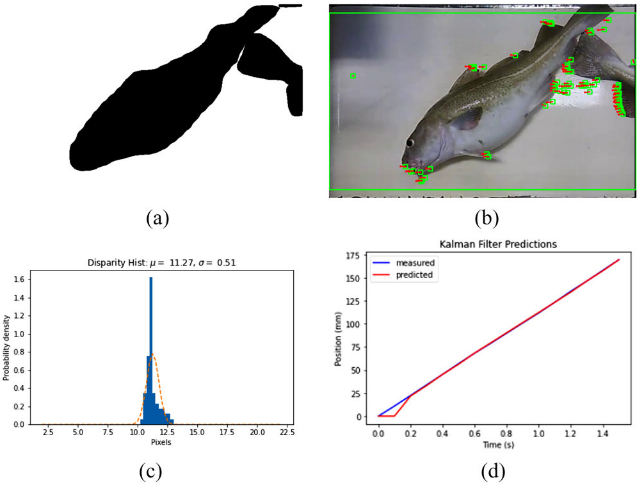


FIGURE 5 Belt motion estimation


(a)

(b)

(c)

(d)

FIGURE 6 Belt motion estimation: (a) Background Mask computed by Mask-RCNN network; (b) Motion vectors due to background point correspondences (A_n, A_{n-1}). Note: ROI excludes the edge of conveyor belt; (c) Histogram of **dx** estimates; (d) Kalman Filter predictions

pipeline is one of a family that search for correspondences in 1D (i.e. along image rows). This technique assumes that the left and right inputs have been transformed so their rows are aligned; a process called rectification. Since the monocular geometry derives stereo disparity from motion induced by the conveyor belt, the stereo pipeline's input covers the conveyor belt region only.

3.4.1 | Video input

Video is recorded by systems supplied by Archipelago Marine Research Ltd. [53] and Anchor Lab. [54]. The specification varies, depending on the supplier. Some installations deploy rolling shutter cameras, others favour global shutter cameras. The frame rate of the recorded footage depends on the system configuration. However, REM systems typically downsample the camera's video stream by dropping frames and overwrite the camera's frame timestamps. We estimate the camera's frame rate using conveyor belt motion cues and resample the stored footage so frames are equispaced in time as this allows us to deploy a textbook Kalman filter to track the position of the belt [40].

3.4.2 | Frame buffer

Video frames are stored in a buffer that is configured as a First In First Out (FIFO) queue (Figure 3). The buffer stores frames, masks, and accumulated belt position. Frames are transformed before they are enqueued, to correct lens distortion, map the image into a rectilinear space, and extract the conveyor belt region (Figure 4).

Rolling shutter distortion is corrected using the approach outlined below and frames are enqueued until the required baseline is achieved (i.e. baseline ≥ target baseline). The incremental conveyor belt motion (dx') is found by computing optical flow from a sparse set of SIFT keypoints located on the surface of the conveyor belt. See Section 3.4.3 for further details.

- **Correct lens distortion**: The surveillance cameras used on board fishing vessels often employ fish-eye lenses to increase the field of view. This introduces severe lens distortion in the image which compromises the search for stereo matches. The radial distortion coefficients necessary to correct lens distortion and intrinsic matrices necessary for transforming from camera coordinates to world coordinates are found by an offline camera calibration procedure [59, 60]. This information is used to undistort the video frames that (after rectification) become the left and right images.
- **Belt extraction**: Once lens distortion has been corrected frames are rectified by finding a homography that corrects any perspective distortion and applies an affine transformation that presents the conveyor belt region in the image frame. This transform is precomputed by interactively identifying the conveyor belt region of interest (corners) in a video of a chess board placed on the conveyor belt. After transformation, the resulting frames are effectively rectified, thereby allowing the search for correspondences to be focused along rows. For further details see French et al. [3].
- **Crop ROI**: A region-of-interest (ROI) that excludes the edges of the conveyor belt is also precomputed (see Figure 6b).
- **Correct rolling shutter distortion**: Some cameras use a horizontal rolling shutter and therefore the scanlines are exposed sequentially. Assuming the frame starts at t_0, the row index of the scanline can be expressed as a function of time by:

$$v_{cam}(t - t_0) = rt - v_0, \tag{5}$$

where r is the rate of rows per microsecond, and v_0 is the index of the first exposed scanline [61]. Meingast et al. [61] develop a general projection equation for a rolling shutter and show how it is affected by different types on camera motion. In our case the camera is fixed. Assuming image motion is constant and only due to the conveyor belt moving in the x-y Cartesian plane (i.e. we assume effects due to disparity are negligible), then pixel coordinates x_r, y_r in a video frame captured with a rolling shutter camera are mapped to x_g, y_g in an equivalent video frame captured with a global shutter camera:

$$\begin{aligned} x_g(\rho,\theta) &= x_r + \rho\frac{\upsilon}{h}\cos\theta \\ y_g(\rho,\theta) &= y_r + \rho\frac{\upsilon}{h}\sin\theta \end{aligned}, \tag{6}$$

where υ is the row index Equation (5), ρ is the distance (no. of pixels) travelled by the conveyor belt in (t = 1/fps s), h is the number of rows in the cropped video frame and θ is the direction of travel.

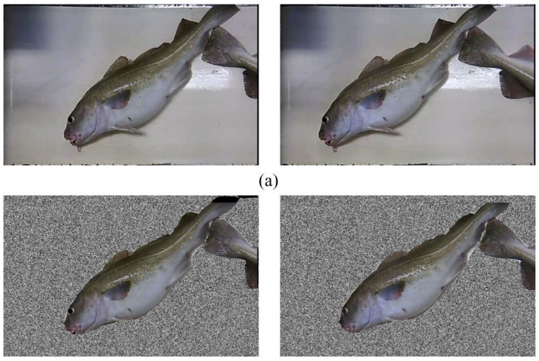
3.4.3 | Measuring the stereo baseline

The distance moved by the conveyor belt between acquisition of the left and right image must be precisely measured (ideally with sub-pixel accuracy) since it determines the stereo baseline. We estimate belt motion by determining correspondence of SIFT interest points [41]. We search for SIFT interest points in an ROI that covers the conveyor belt but excludes foreground objects as these point correspondences include an additional disparity and over represent the true belt motion. We discard any outliers and represent the individual displacements as a vector **dx**. We select the most frequent histogram bin as a global estimate *dx* of the conveyor belt position. These estimates form the input of a Kalman Filter which tracks the belt position in successive video frames [40]. The processing pipeline is depicted in Figure 5.

Let, $\{\mathbf{x}\} = \{\mathbf{x}^f\} \cup \{\mathbf{x}^b\}$, be the set of all interest points in the ROI where $\mathbf{x}^f$ and $\mathbf{x}^b$ represent the interest points that cover foreground and background objects respectively. Considering only the points lying in the background, let $A_n = \{\mathbf{x}_n^b\}$ the set of background interest points in the current frame (f_n) and $A_m = \{\mathbf{x}_m^b\}$ the set of background interest points in the previous frame (f_{n-1}). The segmentation mask used to separate $\mathbf{x}^f$ and $\mathbf{x}^b$ is precomputed using a Mask-RCNN network [21], originally trained for species classification [3]. An example mask is shown in Figure 6a. We use OpenCV's Fast Library for Approximate Nearest Neighbors (FLANN) to identify background SIFT point correspondences. FLANN is a library comprising a collection of algorithms for fast nearest neighbour search that targets large datasets and high dimensional feature spaces. The approach was developed by Muja and Lowe [62] who formalise the nearest neighbor search problem as follows:

"Given a set of points $P = \{p_1, \ldots, p_n\}$ in a vector space X, these points must be preprocessed in such a way that given a new query point $q \in X$, finding the points in P that are nearest to q can be performed efficiently."

To match the SIFT keypoint vectors we configure FLANN for hierarchical k-means clustering. The method clusters the given feature vectors by constructing a hierarchical k-means tree and chooses a cut in the tree that minimizes the cluster's variance. Using this approach we establish pairwise correspondences between SIFT points identified in consecutive frames and compute a set of motion vectors $\mathbf{c}_{mn} = (\mathbf{dx}, \mathbf{dy})$ connecting points x_n^b and x_m^b. Examples of the inlier ($\leq$ 1 Std. Error) motion vectors $\mathbf{c}_{mn}$. are shown in Figure 6b.



(a)

(b)

FIGURE 7 Stereo pre-processor: (a) Input Stereo Pair; (Tb = 0.5 s); (b) Output Stereo Pair showing background replaced with pseudo-random texture to force reference disparity on the conveyor belt

The **dy** component of $\mathbf{c}_{mn}$ is negligible due to rectification and **dx** represents the belt motion. Individual belt motion estimates $h_f(\mathbf{dx})$ are noisy so we build a histogram and select the central value of the most frequent bin to represent the global belt motion *dx* (see Figure 6c).

Finally, a Kalman Filter is deployed to predict the belt position *px* from estimates *dx* and the measurement noise is set by examining the frequency histogram $h_f(\mathbf{dx})$ [40]. Figure 6d illustrates the performance of the tracker.

3.4.4 | Stereo pre-processor

The left and right stereo pair (Figure 7a) can be processed directly by the stereo pipeline, but this is rather inefficient since the search for disparities is dominated by the conveyor belt motion. Additionally, some conveyor belts appear white and featureless, and the lack of texture is very challenging for the stereo block matching network. We address this problem by translating the right frame by a distance equivalent to the baseline, and in-paint the belt surface (background) with a pseudo-random texture to force zero disparity on the belt (Figure 7b).

The input to the preprocessor is a pair of frames acquired at time t_0 and $t_0 + T_b$ from a monocular camera monitoring a conveyor system. T_b is determined by the desired target baseline.

The stereo pre-processor performs the following tasks:

- **Segment foreground** For subsequent steps we need to separate foreground objects and the background belt surface. We reuse the precomputed Mask-RCNN masks (e.g. Figure 6a) for this purpose.
- **Translate right image** We translate the right frame horizontally by υ, where υ represents the sum of incremental belt shifts stored in the frame buffer. Applying a translation reduces the size of the search window (no. of disparities) for the stereo pipeline.
- **Render background** We render the background in both left and right frames in a pseudo-random texture. This forces

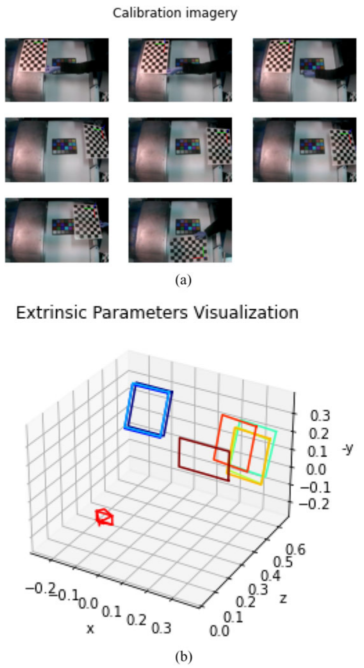


FIGURE 8 Camera extrinsics (MRV Scotia): (a) Calibration imagery; (b) Camera centric extrinsics visualisation

a ‘zero-disparity’ match for the background (i.e. the belt surface).

3.4.5 | Stereo correspondence

We search for correspondences using a feature based approach that employs a Matching Cost - Convolutional Neural Network (MC-CNN) trained on the Middlebury 2014 dataset [32]. The network determines the cost of matching 11x11 pixel patches and forms part of a stereo pipeline proposed by Zbontar et al. [13]. The pipeline comprises MC-CNN, followed by cross-based cost aggregation, semi-global-matching (SGM), interpolation of mis-match and occlusion, sub-pixel enhancement and bilateral filtering. We developed our own Pytorch implementation of MC-CNN (the original implementation is written in Lua) and the other pipeline functions were taken from a GPU library published by Xing Mei et al. [57], called by a Python/C API. We also translated the GPU library into Python to allow us to run on the CPU for debugging and offline testing. Due to the computational complexity of cross-based cost aggregation we usually skip this step when running on the CPU. An example of the input to the stereo pipeline is shown in Figure 7b.

3.4.6 | Stereo post-processor

The post-processor performs two tasks. It adds the baseline disparity to the disparity map (reversing the translation applied by the preprocessor) and transforms the disparity map into a calibrated xyz depth map using the projective transformation matrix **Q**.

$$\mathbf{Q} = \begin{bmatrix} 1 & 0 & 0 & -c_x \\ 0 & 1 & 0 & -c_y \\ 0 & 0 & 0 & f \\ 0 & 0 & -\frac{1}{B} & 0 \end{bmatrix}, \tag{7}$$

where c_x and c_y are the coords. of the camera’s principle point, f represents the focal length and B the length of the baseline.

The viewing geometry can be recovered using calibration imagery (Figure 8). By examining MRV Scotia’s camera extrinsics we can determine that the camera is mounted 1289mm above the conveyor belt. To display a calibrated depth map representation of the aligned image we compute the focal length by rearranging Equation (4).

$$f = \frac{Zd}{B}. \tag{8}$$

Let $Z = Z_b$ represent the distance to the surface of the conveyor belt (recovered from camera extrinsics). Then,

$$f = \frac{Z_b BP}{B} = Z_b P, \tag{9}$$

where P represents the number of pixels per mm in the aligned image. We set $c_x = \frac{w}{2}$ and $c_y = \frac{h}{2}$ where w and h represent the height and width of the aligned image.

3.4.7 | Testing

We produced depth maps and 3D point clouds from video acquired during trials undertaken as part of a preliminary evaluation of REM [2] and Smartfish H2020 [17].

3.5 | Reference system

A laboratory model was constructed to explore the architecture and provide a performance benchmark. The model conveyor system comprised a stepper motor, PVC sheet and rollers cut from plastic tubing. A low cost USB stereo camera (stereo baseline = 62 mm) was mounted 750mm above the conveyor belt. We used FFmpeg [63] to record video and synchronised the left and right cameras based on their frame timestamps. Checking the timestamps and by introducing a ‘flash’ exposure we confirmed that synchronisation remained reasonably stable and frames were acquired at 30 fps. Note: More recent versions of the ELP Stereo Camera provide synchronised outputs. The operational system architecture was modified to allow selection between monocular and binocular camera inputs (Figure 9).

ArUco markers were attached to the conveyor belt (Figure 10) to enable its position to be easily determined without the need for a Kalman Filter [64]. For a conventional stereo

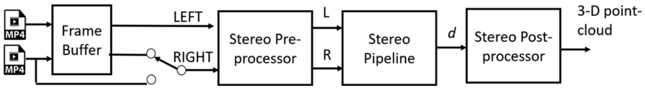


FIGURE 9 Reference system architecture

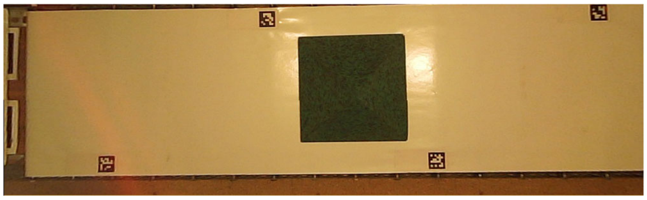


FIGURE 10 Laboratory rig, showing ArUco markers fixed to conveyor belt

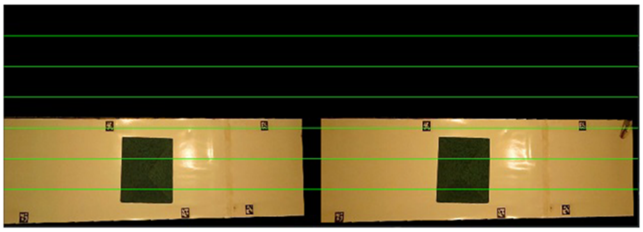


FIGURE 11 Rectified stereo pair from the laboratory rig. Showing segmented conveyor with ArUco markers. Note: Horizontal lines have been added to visualise rectification

camera rectification determines a transformation of each image such that pairs of conjugate epipolar lines become collinear and parallel to the horizontal image axis [65, 66]. The stereo camera was calibrated using the OpenCV library [67]. In addition to calculating lens distortion coefficients, and camera intrinsics, the library also provides a rectification matrix to align the left and right images (e.g. Figure 11) and **Q** matrix (Equation 7) to project the disparity map to 3D world coordinates.

For the laboratory rig we assume only one object is present at a time. This constraint allows us to use a simple foreground segmentation threshold followed by a morphological filter to remove noise.

3.5.1 | Testing

To assess the performance of the reference system we constructed a number of calibration objects. These were printed with a pseudo-random pattern designed to ensure their reflectance functions were approximately lambertian and albedos non-uniform (i.e. rich in non-homogeneous textures). We also built 3D CAD models to provide ground truth (Figure 12).

We acquired images at high definition (HD) resolution (1280 x720 pixels) and standard definition (SD) resolution (640x360 pixels down sampled). To test precision we produced 3D point clouds from 30 frames (1 s) of video acquired with the belt stopped and in motion with the calibration object near the camera's optical centre. We fitted bounding boxes to the 3D point clouds and measured length, width, and height. To test accuracy we registered one of the point clouds (acquired close to the optical centre) with the CAD model (upsampled) and used Open3D method compute_point_cloud_distance to compute the distance from the output point cloud to the target (CAD model ground truth) point cloud. We also analysed the point cloud and estimated x,y,z quantisation.

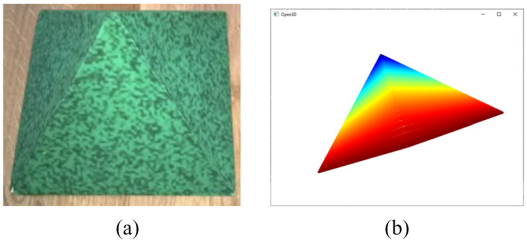


FIGURE 12 (a) Calibration pentahedron. Base = 190x190 mm; height = 140 mm. (b) CAD model

TABLE 1 Reference system: Results from one stereo pair

System config-uration	Base-line (mm)	Voxel size (mm)	BBox (mm)	Misma-tches (pixels)
Binocular (SD)	62.0	1.7x1.7x4.6	190.6x188.0x123.2	477
Binocular (HD)	62.0	0.9x0.9x2.2	189.7x188.0x126.8	9366
Monocular (SD)	62.2	1.7x1.7x4.6	183.6x182.7x122.9	337
Monocular (HD)	62.2	0.9x0.9x2.2	183.6x182.7x122.9	7983

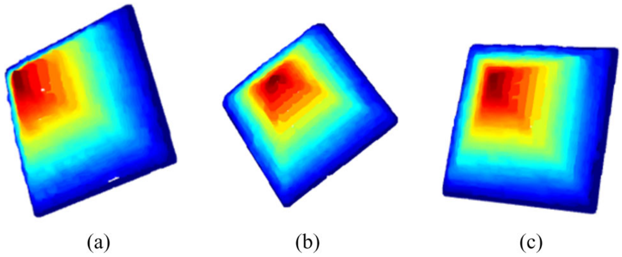


FIGURE 13 3D point clouds rendered as heat colour map.: (a) Binocular stereo (HD), belt stopped; (b) Binoccular stereo (HD), Belt moving; (c) Monocular stereo (HD), Belt moving

4 | RESULTS

4.1 | Reference system

Table 1 summarises the operational parameters of the binocular and monocular systems. A comparison of 3D point clouds and depth maps produced from monocular and binocular video are presented in Figures 13 and 14.

An analysis of the size of bounding boxes enclosing the point clouds in 30 consecutive frames is presented in Figure 15 and the distance between source and target (CAD model) point clouds is shown in Figure 16.

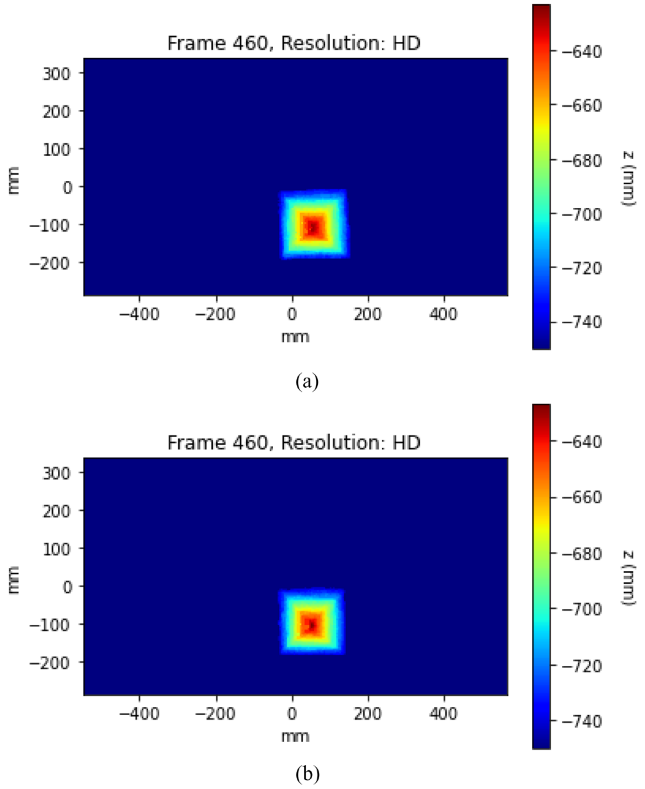


FIGURE 14 Depth maps: (a) Binocular stereo (HD); (b) Monocular stereo (HD)

We compared 3D objects recovered by binocular and motion stereo as follows. First we down-sampled both the point clouds to a voxel size of 2.5 mm and computed the distance between them. We found that 97.5% of the points were coincident and those non-coincident corresponded to outlying points, rendered in red in Figure 17. After removing the outliers we found that there was no difference between the point clouds, that is, all (approx. 10,000) points were coincident (Figure 17c).

4.2 | Operational system

Depth maps obtained from video footage acquired on board MRV Scotia are presented in Figure 18 and rendered as 3D point clouds in Figure 19. Further results are shown in Section A. The throughput of the GPU implementation is approximately 5 fps (1 x NVIDIA Tesla K40).

5 | DISCUSSION

The results demonstrate that MC-CNN trained on the Middlebury stereo data set produces good reconstructions of 3D scenes both in the laboratory and at sea. This is interesting, since the Middlebury training images represent a variety of scenes comprising everyday objects that are very different to the target

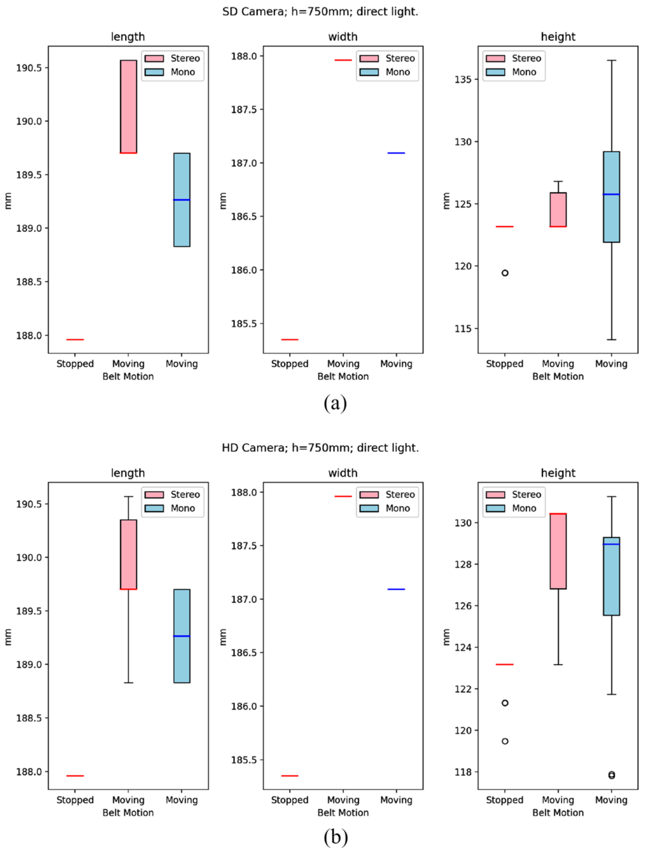


FIGURE 15 Estimates of the calibration object's BBox in 30 stereo frames. (a) Standard definition (SD); (b) High definition (HD). Note: BBox ground truth = 190x190x140 mm

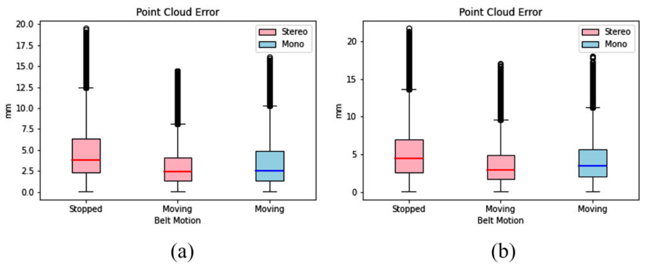


FIGURE 16 Distance between registered 3D point clouds: (a) Standard definition; (b) High definition

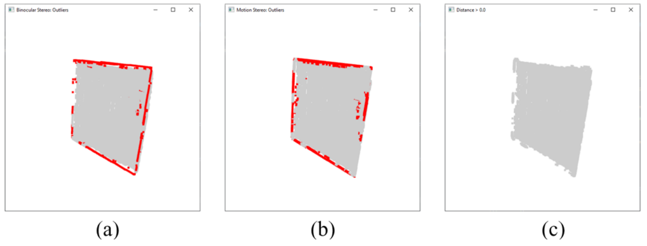
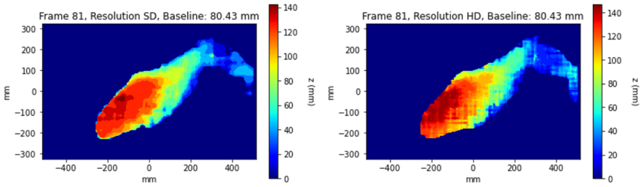
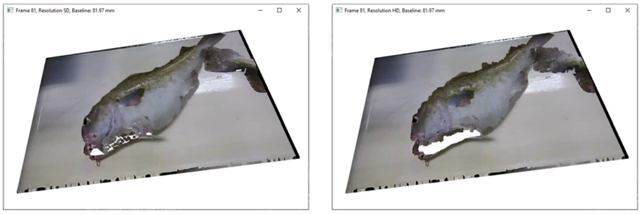


FIGURE 17 Comparison of (HD) point clouds: (a) Binocular stereo outliers (red); (b) Motion stereo outliers (red); (c) Matching points (grey) after outliers have been removed



(a) (b)

FIGURE 18 Depth maps: (a) Standard Definition; (b) High Definition. Note: z (mm) relative to conveyor belt


Frame 81, Resolution SD, Baseline: 81.97 mm
Frame 81, Resolution HD, Baseline: 81.97 mm

(a) (b)

FIGURE 19 Point clouds Rendered with image texture): (a) Standard Definition (SD); (b) High Definition (HD)

domain. One of the disadvantages of neural networks is that the features encoded by the network are somewhat opaque, so we cannot claim that MC-CNN learns mutual information, image gradients, colour difference, etc. However, the network appears to learn a set of universal features (within a 11x11 window) that enables it to match stereo imagery captured in a range of scenarios. Although we did not confirm experimentally that the network is robust to changes in illumination, since MC-CNN is ranked higher in the Middlebury evaluation than other approaches that do claim illuminant invariance, we might expect it to perform well in this regard.

5.1 | Laboratory reference

A striking feature in Table 1 is the high number of mismatched pixels at HD resolution compared to SD resolution for both binocular and monocular systems. This illustrates that block matching at HD is much more sensitive to errors in correcting radial lens distortion. The mismatches are corrected by an in-painting algorithm and there is little difference in the visual appearance of 3D point clouds and depth maps presented in Figures 13 and 14.

Figure 15 confirms that with no conveyor belt motion, the binocular camera recovers the pentahedron's bounding box (BBox) size with high precision. This represents a benchmark level of performance, such as could be expected from synchronised binocular stereo head employing a global shutter monitoring a belt in motion. When the conveyor belt moves the noise due to the lack of synchronisation and motion blur (the rolling shutter integration period is approx. 1/30 s) are introduced along image rows and this is reflected in the BBox length and height estimates. The BBox width is not significantly affected by the horizontal shutter or motion artifacts. The monocular and binocular systems perform similarly however heights estimated by the monocular system tend to be noisier. This is most likely due firstly to different camera intrinsics, that is, the binocular system stereo forms images from the left, right camera and the monocular system forms images from the left camera only, and secondly reflects some uncertainly in estimating the effective baseline. All estimates tend to under represent the true BBox size as the stereo pipeline matches patches of size 11x11 pixels.

Figure 16 compares the accuracy of the systems and confirms the HD imagery delivers better estimates than SD, although not by the margin one might expect given the voxel sizes reported in Table 1. This is probably due to the larger number of mismatched pixels at HD resolution. The significant outliers are mainly due to inaccuracies in height estimates. Figure 16 shows these could reach a maximum of 18 mm.

Comparing binocular and motion stereo point clouds we find 97.5% of points are coincident with errors mostly confined to the interface between the belt and the base of the polyhedron (Figure 17). These mismatches occur at a transition between two similar pseudo-random textures; rendered onto the conveyor belt and printed on the polyhedron. This presents challenges for the stereo pipeline and it is not surprising that mismatches occur at this interface. Fortunately, these points can be identified as outliers, and after removing them we find there to be no difference between point clouds from binocular and motion stereo.

5.2 | MRV Scotia

We have no ground-truth for depth maps derived from MRV Scotia's video footage. A possible source of inaccuracy in estimating depth is in determining the baseline. This becomes increasingly difficult as the density of fish carried by the conveyor system increases since there is less exposed belt. The REM CCTV camera was installed before this work was envisaged and is located 1289 mm above the conveyor. The quantisation in z is approx. 8mm (SD resolution) at an equivalent baseline of 80 mm. Figure A1 illustrates some challenges for the system. Part of the fish rendered in Figure A1d is missing due to an error with the mask, possibly caused by the region of shadow. The depth map for the smallest fish in Figure A1f is corrupted, probably due to the lack of texture on the underside of the fish. Figure A2 illustrates noise due to inconsistencies in the baseline estimates which manifest as a lack of precision in the depth maps.

5.3 | North sea trawler

Figure A3 illustrates some challenges with motion stereo in an operational setting. In this case the camera is mounted 1278 mm above the conveyor belt. A crew member is gutting fish and a gloved hand can be seen in the right of the frame. Splatter from this work has fouled the dome protecting the camera lens and parts of the image appears blurred. The conveyor belt is moving debris and discarded fish to the left; on this side the camera dome is cleaner, objects appear sharper and we have successfully recovered the depth.

We have yet to acquire ground truth depth from fish conveyor systems or analyse sufficient data to draw conclusions regarding the accuracy of biomass estimates using Motion Stereo. However, we already use a generic geometric model to provide estimates of depth to generate shadows in synthesised imagery that is used to augmented training data. We expect that depth estimates produced by Motion Stereo will represent an improvement over the generic models we currently employ.

6 | CONCLUSION

We have demonstrated the efficacy of a system that computes depth from a sequence of monocular images obtained from REM CCTV monitoring fish in transit on a conveyor belt. The motion constraint imposed by the conveyor belt allowed us to recover depth by Motion Stereo. In this case the stereo baseline is determined by the distance travelled by the conveyor belt. We addressed some practical challenges such as correction of motion artifacts due to the camera's rolling shutter, and adopted MC-CNN stereo matching due to Zbontar and LeCun [13] to produce disparity maps. By observing a calibration pentahedron placed on a model conveyor system constructed in the laboratory with a stereo camera at a distance of 0.75 m above, we compared 3D point clouds rendered by binocular and monocular stereo with a ground truth CAD model. We fixed ArUco markers to our laboratory rig to help accurately determine belt motion and found that Motion Stereo and conventional binocular stereo performed equivalently. Comparing the recovered point cloud with ground truth gave a median error of $<$ 5 mm and the maximum error of 15 mm. We also compared the Binocular and Motion Stereo point clouds directly and found that after removing outlying points ($>$2 Std. Error) there was no difference between the 3D point clouds. We tested with standard and high definition imagery and found that with HD a significant number of mismatched pixels were reported by the stereo pipeline. We believe this to be due to inaccuracy in correcting for radial lens distortion. The relatively large maximum error is due to the 11x11 patch size used by MC-CNN. We estimated precision to be $<$ 4 mm (interquartile interval) by analysing 30 consecutive video frames acquired near the camera's optical centre. We also presented depth maps rendered from operational video acquired from a REM CCTV system installed on board Marine Scotland's research vessel MRV Scotia. This presented further challenges as we found the video timestamps had been overwritten when the REM video was recorded. Recovering belt motion was more challenging due to dropped frames and the absence of ArUco markers. In this case we recovered the conveyor belt motion by tracking SIFT interest points with a Kalman Filter. The depth maps obtained from the operational system are quite noisy but sufficient for some tasks such as applying training data augmentation for training of further CNN models. In future work we plan to use this data to augment the species classifier training set and investigate other machine learning techniques for estimating depth from monocular imagery.

AUTHOR CONTRIBUTIONS

Mark Fisher: Funding acquisition, software, writing - original draft. Geoffrey French: Software. Artjoms Gorpincenko: Data curation. Helen Holah: Investigation, validation, writing - review and editing. Lauren Clayton: Investigation, validation, writing - review and editing. Rebecca Skirrow: Investigation, validation, writing - review and editing. Michal Mackiewicz: Funding acquisition, project administration, resources, writing - review and editing.

ACKNOWLEDGEMENTS

This work has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 773521. Our results were produced using the University of East Anglia's High Performance Linux Compute Cluster and we would like to thank Dr. Jacob Newman for his help configuring the software modules and packages that we required.

CONFLICT OF INTEREST

All authors declare that they have no conflicts of interest.

DATA AVAILABILITY STATEMENT

The data acquired in the laboratory that support the findings of this study are available on request from the corresponding author. The data acquired operationally at sea are not publicly available due to restrictions, for example, their containing information that could compromise the privacy of research participants.

ORCID

Mark Fisher https://orcid.org/0000-0002-8651-543X
Helen Holah https://orcid.org/0000-0003-3629-3303

REFERENCES

1. van Helmond, A.T.M., Mortensen, L.O., Plet.Hansen, K.S., Ulrich, C., Needle, C.L., Oesterwind, D., et al.: Electronic monitoring in fisheries: Lessons from global experiences and future opportunities. Fish Fisheries 21(1), 162–189 (2020)
2. Needle, C.L., Dinsdale, R., Buch, T.B., Catarino, R.M.D., Drewery, J., Butler, N.: Scottish science applications of Remote Electronic Monitoring. ICES J. Mar. Sci. 72(4), 1214–1229 (2014)
3. French, G., Mackiewicz, M., Fisher, M., Holah, H., Kilburn, R., Campbell, N., et al.: Deep neural networks for analysis of fisheries surveillance video and automated monitoring of fish discards. ICES J. Mar. Sci. 08, (2019)
4. French, G., Fisher, M., Mackiewicz, M., Needle, C.: Convolutional neural networks for counting fish in fisheries surveillance video. In: Amaral, T., Matthews, S., Plötz, T., McKenna, S., Fisher, R. (eds.) Proceedings of the Machine Vision of Animals and their Behaviour (MVAB), pp. 7.1–7.10. BMVA Press, London (2015).
5. Nevatia, R.: Depth measurement by motion stereo. Comp Graph Image Process 5(2), 203–214 (1976)
6. Ens, J., Li, Z.N.: Real-time motion stereo on sfu pyramid. Real-Time Imag. 1(6), 385–396 (1995)
7. Hashim, H.S., Abdullah, S.N.H.S., Prabuwono, A.S.: Automated visual inspection for metal parts based on morphology and fuzzy rules. In: 2010 International Conference on Computer Applications and Industrial Electronics, pp. 527–531. IEEE, Piscataway (2010)

8. Alper Selver, M., Akay, O., Alim, F., Bardakçı, S., Ölmez, M.: An automated industrial conveyor belt system using image processing and hierarchical clustering for classifying marble slabs. Rob. Comput. Integr. Manuf. 27(1), 164–176 (2011)
9. Seredkin, A.V., Tokarev, M.P., Plohih, I.A., Gobyzov, O.A., Markovich, D.M.: Development of a method of detection and classification of waste objects on a conveyor for a robotic sorting system. J. Phys. Conf. Ser. 1359, 012127 (2019)
10. Gundupalli, S.P., Hait, S., Thakur, A.: A review on automated sorting of source-separated municipal solid waste for recycling. Waste Manage. 60, 56–74 (2017). Special Thematic Issue: Urban Mining and Circular Economy.
11. Dhakshina Kumar, S., Esakkirajan, S., Bama, S., Keerthiveena, B.: A microcontroller based machine vision approach for tomato grading and sorting using svm classifier. Microprocess. Microsyst. 76, 103090 (2020)
12. Nashat, S., Abdullah, A., Aramvith, S., Abdullah, M.Z.: Support vector machine approach to real-time inspection of biscuits on moving conveyor belt. Comput. Electron. Agric. 75(1), 147–158 (2011)
13. Žbontar, J., LeCun, Y.: Stereo matching by training a convolutional neural network to compare image patches. J. Mach. Learn. Res. 17(1), 2287–2318 (2016)
14. Mathiassen, J.R., Misimi, E., Toldnes, B., Bondø, M., Østvik, S.O.: High-speed weight estimation of whole herring (clupea harengus) using 3d machine vision. J. Food Sci. 76(6), E458–E464 (2011)
15. Hao, M., Yu, H., Li, D.: The measurement of fish size by machine vision - a review. In: Li, D., Li, Z. (eds.) Computer and Computing Technologies in Agriculture IX. CCTA 2015. IFIP Advances in Information and Communication Technology, vol. 479. Springer, Cham (2016)
16. Zion, B.: The use of computer vision technologies in aquaculture – a review. Comput. Electron. Agric. 88, 125–132 (2012)
17. Cordis: Smart fisheries technologies for an efficient, compliant and environmentally friendly fishing sector. https://cordis.europa.eu/project/id/773521. Accessed 15 October 2021
18. Coastal Resources Management Group.: Automated shellfish species, size and sex identification system (AS3ID) (2020-2021). https://crmg.st-andrews.ac.uk/current-projects/automated-shellfish-species-size-and-sex-identification-system-as3id-2020-2021/. Accessed 02 November 2021
19. Tseng, C.H., Kuo, Y.F.: Detecting and counting harvested fish and identifying fish types in electronic monitoring system videos using deep convolutional neural networks. ICES J. Mar. Sci. 77(4), 1367–1378 (2020)
20. Álvarez Ellacuría, A., Palmer, M., Catalán, I.A., Lisani, J.L.: Image-based, unsupervised estimation of fish size from commercial landings using deep learning. ICES J. Mar. Sci. 77(4), 1330–1339 (2019)
21. He, K., Gkioxari, G., Dollar, P., Girshick, R.: Mask r-cnn. In: Proceedings of the IEEE International Conference on Computer Vision (ICCV). IEEE, Piscataway (2017)
22. Garcia, R., Prados, R., Quintana, J., Tempelaar, A., Gracias, N., Rosen, S., et al.: Automatic segmentation of fish using deep learning with application to fish size measurement. ICES J. Mar. Sci. 77(4), 1354–1366 (2019)
23. Petrellis, N.: Measurement of fish morphological features through image processing and deep learning techniques. Appl. Sci. 11(10), 4416 (2021)
24. Rosen, S., Holst, J.C.: Deepvision in-trawl imaging: Sampling the water column in four dimensions. Fish. Res. 148, 64–73 (2013)
25. Özyeşil, O., Voroninski, V., Basri, R., Singer, A.: A survey of structure from motion.. Acta Numerica 26, 305–364 (2017)
26. Makhubela, J.K., Zuva, T., Agunbiade, O.Y.: A review on vision simultaneous localization and mapping (vslam). In: 2018 International Conference on Intelligent and Innovative Computing Applications (ICONIC), pp. 1–5. IEEE, Piscataway (2018)
27. Longuet.Higgins, H.C.: A computer algorithm for reconstructing a scene from two projections. Nature 293, 133–135 (1981)
28. Wang, H., Zhou, W., Zhang, X., Lou, X.: A block patchmatch-based energy-resource efficient stereo matching processor on FPGA. IEEE Trans. Circuits Syst. I Regul. Pap. (2022)
29. Itseez.: Open source computer vision library (2015). https://github.com/itseez/opencv
30. Trucco, E., Verri, A.: Introductory Techniques for 3-D Computer Vision. Prentice Hall PTR, Upper Saddle River (1998)
31. Scharstein, D., Szeliski, R.: A taxonomy and evaluation of dense two-frame stereo correspondence algorithms. Int. J. Comput. Vision 47(1), 7–42 (2002)
32. Scharstein, D., Hirschmüller, H., Kitajima, Y., Krathwohl, G., Nesic, N., Wang, X., et al.: High-resolution stereo datasets with subpixel-accurate ground truth. In: German Conference on Pattern Recognition (GCPR 2014). Springer, Cham (2014)
33. Geiger, A., Lenz, P., Stiller, C., Urtasun, R.: Vision meets robotics: The kitti dataset. Int. J. Rob. Res. 32(11), 1231–1237 (2013)
34. Middlebury.: Middlebury stereo evaluation - Version 3. https://vision.middlebury.edu/stereo/eval3/. Accessed 3 August 2022
35. Geiger, A., Lenz, P., Stiller, C., Urtasun, R.: Kitti stereo. http://www.cvlibs.net/datasets/kitti/eval_stereo.php. Accessed 3 August 2022
36. Hirschmuller, H.: Stereo processing by semiglobal matching and mutual information. IEEE Trans. Pattern Anal. Mach. Intell. 30(2), 328–341 (2008)
37. Deori, B., Thounaojam, D.M.: A survey on moving object tracking in video. Int. J. Inf. Theor. 3(3), 31–46 (2014)
38. Ojha, S., Sakhare, S.: Image processing techniques for object tracking in video surveillance- a survey. In: 2015 International Conference on Pervasive Computing (ICPC), pp. 1–6. IEEE, Piscataway (2015)
39. Mishra, P.K., Saroha, G.P.: A study on video surveillance system for object detection and tracking. In: 2016 3rd International Conference on Computing for Sustainable Global Development (INDIACom), pp. 221–226. IEEE, Piscataway (2016)
40. Kalman, R.E.: A new approach to linear filtering and prediction problems. ASME J. Basic Eng. 82, 35–45 (1960)
41. Lowe, D.G.: Distinctive image features from scale-invariant keypoints. Int. J. Comput. Vision 60, 91–110 (2004)
42. Song, D., Zhao, B., Tang, L.: A tracking algorithm based on sift and kalman filter. In: 2012 International Conference on Computer Application and System Modeling, pp. 1563–1566. IEEE, Piscataway (2012)
43. Ha, S.W., Moon, Y.H.: Object tracking using sift and kalman filter. In: Proceedings of the International Conference on Image Processing, Computer Vision, and Pattern Recognition (IPCV), p. 1. The Steering Committee of The World Congress in Computer Science, Computer Science Computer Engineering and Applied Computing WorldComp (2014)
44. Mirunalini, P., Jaisakthi, S., Sujana, R.: Tracking of object in occluded and non-occluded environment using sift and kalman filter. In: TENCON 2017-2017 IEEE Region 10 Conference, pp. 1290–1295. IEEE, Piscataway (2017)
45. Geyer, C., Meingast, M., Sastry, S.: Geometric models of rolling-shutter cameras. 6th OmniVis WS 1(4), (2005)
46. Bradley, D., Atcheson, B., Ihrke, I., Heidrich, W.: Synchronization and rolling shutter compensation for consumer video camera arrays. In: 2009 IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops, pp. 1–8. IEEE, Piscataway (2009)
47. Hedborg, J., Ringaby, E., Forssén, P.E., Felsberg, M.: Structure and motion estimation from rolling shutter video. In: 2011 IEEE International Conference on Computer Vision Workshops (ICCV Workshops), pp. 17–23. IEEE, Piscataway (2011)
48. Hedborg, J., Forssén, P.E., Felsberg, M., Ringaby, E.: Rolling shutter bundle adjustment. In: 2012 IEEE Conference on Computer Vision and Pattern Recognition, pp. 1434–1441. IEEE, Piscataway (2012)
49. Saurer, O., Koser, K., Bouguet, J.Y., Pollefeys, M.: Rolling shutter stereo. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 465–472. IEEE, Piscataway (2013)
50. Kim, J.H., Cadena, C., Reid, I.: Direct semi-dense slam for rolling shutter cameras. In: 2016 IEEE International Conference on Robotics and Automation (ICRA), pp. 1308–1315. IEEE, Piscataway (2016)
51. Im, S., Ha, H., Choe, G., Jeon, H.G., Joo, K., Kweon, I.S.: Accurate 3d reconstruction from small motion clip for rolling shutter cameras. IEEE Trans. Pattern Anal. Mach. Intell. 41(4), 775–787 (2019)
52. Ailipu Technology Ltd.: Elp 2mp 1080p ar0330 mjpeg 30fps mini webcam dual lens micro usb camera. http://www.webcamerausb.com/. Accessed 15 October 2021

53. Archipelago Marine Research Ltd.: Fisheries services. https://www.archipelago.ca/. Accessed 15 October 2021
54. Anchor Lab.: Black box video. http://www.anchorlab.net/. Accessed 15 October 2021
55. Paszke, A., Gross, S., Chintala, S., Chanan, G., Yang, E., DeVito, Z., et al.: Automatic differentiation in pytorch. In: NIPS-W. NIPS Foundation, San Diego (2017)
56. Zhou, Q.Y., Park, J., Koltun, V.: Open3d: A modern library for 3d data processing. arXiv abs/1801.09847 (2018)
57. Mei, X., Sun, X., Zhou, M., Jiao, S., Wang, H., Zhang, X.: On building an accurate stereo matching system on graphics hardware. In: 2011 IEEE International Conference on Computer Vision Workshops (ICCV Workshops), pp. 467–474. IEEE, Piscataway (2011)
58. Hartley, R., Zisserman, A.: Multiple View Geometry in Computer Vision. 2nd ed., Cambridge University Press, Cambridge (2004)
59. D'Emilia, G., Gasbarro, D.D.: Review of techniques for 2d camera calibration suitable for industrial vision systems. J. Phys.: Conf. Ser. 841, 012030 (2017)
60. OpenCV.: Camera calibration. https://docs.opencv.org/master/dc/dbb/tutorial_py_calibration.html. Accessed 19 October 2021
61. Meingast, M., Geyer, C., Sastry, S.: Geometric models of rolling-shutter cameras. CoRR abs/cs/0503076 (2005)
62. Muja, M., Lowe, D.G.: Fast approximate nearest neighbors with automatic algorithm configuration. VISAPP (1) 2(331-340), 2 (2009)
63. Tomar, S.: Converting video formats with ffmpeg. Linux J. 2006(146), 10 (2006)
64. Garrido-Jurado, S., Muñoz-Salinas, R., Madrid-Cuevas, F.J., Marín-Jiménez, M.J.: Automatic generation and detection of highly reliable fiducial markers under occlusion. Pattern Recognit. 47(6), 2280–2292 (2014)
65. Ayache, N., Hansen, C.: Rectification of images for binocular and trinocular stereovision. In: [1988 Proceedings] 9th International Conference on Pattern Recognition, vol. 1, pp. 11–16. IEEE, Piscataway (1988)
66. Papadimitriou, D.V., Dennis, T.J.: Epipolar line estimation and rectification for stereo image pairs. IEEE Trans. Image Process. 5(4), 672–676 (1996)
67. OpenCV.: Camera calibration and 3-D reconstruction. https://docs.opencv.org/4.5.3/d9/d0c/group__calib3d.html. Accessed 19 October 2021

How to cite this article: Fisher, M., French, G., Gorpincenko, A., Holah, H., Clayton, L., Skirrow, R., Mackiewicz, M.: Motion stereo at sea: Dense 3D reconstruction from image sequences monitoring conveyor systems on board fishing vessels. IET Image Process. 1–13 (2022). https://doi.org/10.1049/ipr2.12636

APPENDICES A

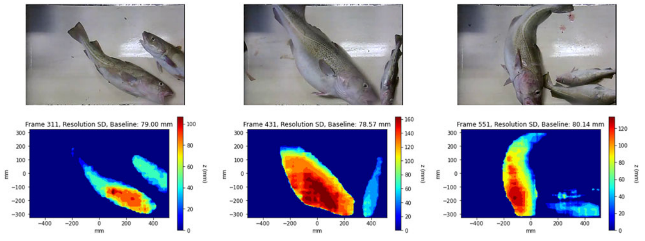


FIGURE A1 Further examples of depth maps from MRV Scotia

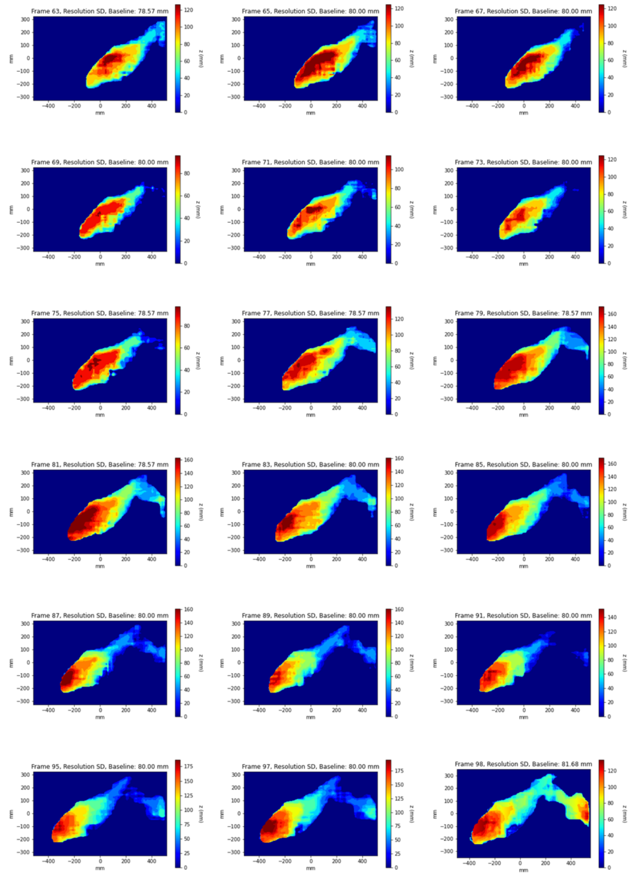
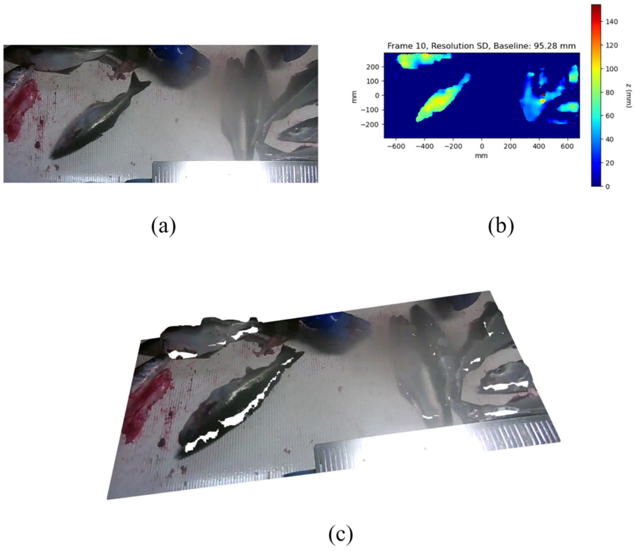


FIGURE A2 Montage of depth maps from 1.8 s clip of video acquired on board MRV Scotia (SD Resolution)



(a)

(b)

(c)

FIGURE A3 Depth map rendered from video footage acquired from a North Sea trawler as part of Marine Scotland's REM trial [2] (SD Resolution). (a) Left image; (b) Depth map; (c) 3D reconstruction