



Kent Academic Repository

Stamoulakatos, Anastasios, Cardona, Javier, McCaig, Chris, Murray, David, Filius, Hein, Atkinson, Robert, Bellekens, Xavier, Michie, Craig, Andonovic, Ivan, Lazaridis, Pavlos and others (2020) *Automatic Annotation of Subsea Pipelines using Deep Learning*. *Sensors*, 20 (3). ISSN 1424-8220.

Downloaded from

<https://kar.kent.ac.uk/79935/> The University of Kent's Academic Repository KAR

The version of record is available from

<https://doi.org/10.3390/s20030674>

This document version

Author's Accepted Manuscript

DOI for this version

Licence for this version

UNSPECIFIED

Additional information

Versions of research works

Versions of Record

If this version is the version of record, it is the same as the published version available on the publisher's web site. Cite as the published version.

Author Accepted Manuscripts













If this document is identified as the Author Accepted Manuscript it is the version after peer review but before type setting, copy editing or publisher branding. Cite as Surname, Initial. (Year) 'Title of article'. To be published in *Title of Journal*, Volume and issue numbers [peer-reviewed accepted version]. Available at: DOI or URL (Accessed: date).

Enquiries

If you have questions about this document contact ResearchSupport@kent.ac.uk. Please include the URL of the record in KAR. If you believe that your, or a third party's rights have been compromised through this document please see our [Take Down policy](https://www.kent.ac.uk/guides/kar-the-kent-academic-repository#policies) (available from <https://www.kent.ac.uk/guides/kar-the-kent-academic-repository#policies>).

Article

Automatic Annotation of Subsea Pipelines using Deep Learning

Anastasios Stamoulakatos ^{1,*}, Javier Cardona ¹, Chris McCaig ¹, David Murray ², Hein Filius ², Robert Atkinson ¹, Xavier Bellekens ¹, Craig Michie ¹, Ivan Andonovic ¹, Pavlos Lazaridis ³, Andrew Hamilton ¹, Md. Moinul Hossain ⁴, Gaetano Di Caterina ¹, Christos Tachtatzis ¹

¹ Department of Electronic and Electrical Engineering, University of Strathclyde Glasgow, UK

² N-Sea, Zierikzee, Netherlands

³ Department of Engineering and Technology School of Computing and Engineering, Huddersfield, UK

⁴ School of Engineering and Digital Arts, University of Kent, Canterbury, Kent, UK

* Correspondence: anastasios.stamoulakatos@strath.ac.uk; Tel.: +44-141-444-7324

Version January 22, 2020 submitted to Sensors

Abstract: Regulatory requirements for sub-sea oil and gas operators mandates the frequent inspection of pipeline assets to ensure that their degradation and damage are maintained at acceptable levels. The inspection process is usually sub-contracted to surveyors who utilise sub-sea Remotely Operated Vehicles (ROVs), launched from a surface vessel and piloted over the pipeline. ROVs capture data from various sensors/instruments which are subsequently reviewed and interpreted by human operators, creating a log of event annotations; a slow, labour-intensive and costly process. The paper presents an automatic image annotation framework that identifies/classifies key events of interest in the video footage viz. exposure, burial, field joints, anodes and free spans. The reported methodology utilises transfer learning with a Deep Convolutional Neural Network (ResNet-50), fine-tuned on real-life, representative data from challenging sub-sea environments with low lighting conditions, sand agitation, sea-life and vegetation. The network outputs are configured to perform multi-label image classifications for the critical events. The annotation performance varies between 95.1% and 99.7% in terms of accuracy and 90.4% and 99.4% in terms of F1-Score depending on event type. The performance results are on a per-frame basis and corroborate the potential of the algorithm to be the foundation for an intelligent decision support framework that automates the annotation process. The solution can execute annotations in real-time and is significantly more cost-effective than human-only approaches.

Keywords: Visual Inspection; Subsea Pipeline Survey; Multi-Label Image Classification; Deep Learning; Transfer Learning

1. Introduction

Oil and Gas operators are governed by regulations which mandate the frequent visual inspections of sub-sea pipelines and platforms in order to assess the condition and risks to these assets. In a typical inspection, a surface vessel deploys a Remote Operating Vehicle (ROV) which is piloted over the pipeline, collecting survey data from multiple sensors/instruments. A typical survey data set comprises: 1) video footage recorded from three camera angles (left/port, centre and right/starboard), 2) Inertial Measurement Unit (IMU) data to capture the orientation of the ROV, 3) multi-beam echo sounder data to map the seabed surface and 4) magnetic pipe-tracker to record the pipe location when it is buried below the seabed.

29 During the inspection, a Data Coordinator, on board the surface vessel, provides real-time
30 commentary on survey data and produces initial annotations, identifying events such as pipeline
31 exposure, burial, field joints, anodes, free spans and boulders. The annotation process is prone to
32 human error [1] as Data Coordinators become fatigued and distracted, leading to missed events or
33 incorrect labelling. After these initial annotations, the video and commentary are subject to Quality
34 Control (QC), either while the survey is ongoing or once completed, creating a bottleneck in the speed
35 of processing and reporting. Furthermore, the speed at which the ROV is piloted is limited by the rate
36 the human can vocalise the presence of an event on audio commentary rather than a limitation of the
37 craft. Automating the survey process enables more consistent, accurate and quicker inspections, while
38 reducing the presence of staff offshore and the concomitant cost and safety risks.

39 Various vision-based techniques proposed by the Autonomous Underwater Vehicle (AUV)
40 navigation community have primarily focused on pipeline tracking, however, they do not detect
41 and annotate events of interest. Jacobi *et al.* [2,3], proposed a pipeline tracking method for AUV
42 guidance through the fusion of optical, magnetic and acoustic sensors applied on simulated pipeline
43 data. Narimani *et al.* [4] proposed a pipeline and cable tracking technique to improve vehicle navigation
44 by converting the images to grey-scale and applying the Hough transformation to determine the angle
45 between vehicle and pipeline; subsequently used as a reference to an adaptive sliding mode controller.
46 Zingaretti *et al.* [1] developed a real-time vision-based detection system [5] for underwater pipelines
47 using edge-based image processing to detect pipeline contours and a Kalman filter that utilises the
48 navigation data to reduce the effect of disturbances created by motion. Similarly, Ortiz *et al.* [6]
49 identified cable contours, in tandem with a linear Kalman filter to predict the contours in the following
50 frame. The same authors presented an alternative approach for tracking using Particle Filters [7] tested
51 with footage obtained in a water tank. Asif *et al.* [8] utilised the Bresenham line algorithm to detect
52 noise-free pipeline boundaries and B-Spline to detect active contours subsequently tracked using a
53 Kalman filter.

54 Sub-sea video footage is particularly challenging to process due to reduced contrast, the presence
55 of suspended particles in the water (e.g. sand, algae), and highly variable illumination. Traditional
56 image processing approaches such as contour determination and their variants, although suitable
57 to localise the edges of the pipeline, require significant feature engineering to detect events such as
58 field joints, free spans and anodes. Sea-life, marine growth, seabed settlements, auxiliary structural
59 elements, breaks on the external pipeline sheathing and alien objects near the pipe are possible
60 sources of confusion in the determination of pipeline contours. Furthermore, it is unclear how these
61 algorithms perform in the absence of the pipeline (when the pipe is buried) or on changes in position
62 and orientation as the ROV maneuvers, both of which result in significant variations of the event
63 appearance in the image plane.

64 Recently, deep learning approaches have been applied with a similar goal within the power line
65 inspection industry [9–12]. Nguyen *et al.* [9] conducted a review on vision-based approaches for power
66 line inspection and the potential role of deep learning. Zhang *et al.* [10] detected electricity poles in
67 Google Street View Imagery using RetinaNet trained with 1,000 annotated images. Jalil *et al.* [11],
68 utilised Faster-RCNN [13] to detect insulators in drone imagery. Miao *et al.* [12] implemented a bespoke
69 Single Shot Detector with MobileNet as the backbone to detect insulators. Various applications can
70 also be found for sub-sea imaging. Bonnin-Pascual and Ortiz [14] presented a framework for defect
71 detection on vessels. The approach pre-computed and combined a range of multi-scale normalized
72 feature maps with the use of Gaussian and Gabor pyramid filters. The framework was successfully
73 tested on image mosaics during vessel inspection campaigns. Bonin-Font *et al.* [15] performed detection,
74 mapping and quantification of *Posidonia Oceanica*. After initially extracting 168 features from images
75 using a range of kernels and the gray-level co-occurrence matrix, 14 classifiers were trained and
76 compared. Principal Component Analysis (PCA) was applied on the best performing model (Logistic
77 Model Trees) to select the 25 more relevant features and retrain the classifier.

78 In a continuation of this work, Martin-Abadal *et al.* [16] created a framework for the semantic
 79 segmentation of *Posidonia Oceanica*. A Deep Fully Convolutional Network was established by
 80 VGG16, pre-trained on ImageNet as an encoder, FCN8 as a decoder with Gaussian initialization of its
 81 parameters and hyper-parameter tuning. Their model was successfully implemented on a Turbot AUV
 82 for online segmentation of meadows.

83 In terms of pipeline inspection, Petraglia *et al.* [17], after initially pre-processing the RGB images,
 84 detected pipeline boundaries by firstly filtering edges through Non-Maximum Suppression (NMS)
 85 to eliminate horizontal line segments followed by Random Sample Consensus (RANSAC) and Total
 86 Least Square (TLS) to group segments. The authors compared two Neural Network (NN) architectures
 87 to classify four type of events: inner coating exposure, algae, flange and concrete blankets. The first
 88 NN architecture utilises two convolutional and three fully connected layers, trained on segmented
 89 pipelines from the pre-processed images. The second architecture adopted a Multilayer Perceptron
 90 (MLP) with a single hidden layer, trained on features extracted from 3-level Wavelet decomposition.
 91 The mean and the variance of the wavelet coefficients at each level are then used as features for the
 92 neural network, except for the mean of the level-1 low-low coefficients, since the window mean is zero.
 93 This feature extraction results in 23 input features from each window. Results led to the conclusion
 94 that the convolutional neural network outperforms the MLP, without any need for manual feature
 95 extraction.

R3.1

96 In this work, transfer learning is harnessed to train a Deep Convolutional Neural Network on
 97 raw images of sub-sea pipeline surveys to automatically classify five events (exposure, burial, free
 98 span, field joint, anode). The performance evaluation of the proposed framework is conducted on data
 99 sets from survey video data obtained from an operational class ROV. The network is configured to
 100 perform multi-label image classification which identifies multiple concurrent events in a single frame
 101 (for example, exposure and field joint). Data augmentation is used to enhance further the training data
 102 sets, facilitating the treatment of the variability embedded within sub-sea images owing to challenges
 103 created by dynamic ROV motion, brightness and contrast. Multiple ResNet models of varying depth
 104 have been trialed and a ResNet-50 architecture was selected because it balances the trade-off between
 105 performance and computation inference time. The ResNet-50 performance yields a high overall Exact
 106 Match Ratio and F1-Score of 91.9% and 96.6% respectively on per single frame basis.

R3.3 & R3.5

107 2. Materials and Methods

108 Data sets from two North Sea surveys conducted in 2012 and 2016 covering 201 kilometres and
 109 58 kilometres, respectively were utilised in the development of the automatic annotation system. Each
 110 survey recorded three synchronised video feeds (left, centre and right) of the pipeline at 25 frames
 111 per second. For the purposes of the development, the centre camera video only was processed for
 112 the following events of interest; examples for various lighting conditions, seabed characteristics and
 113 parasites are shown in Figure 1:

R1.3

see Fig. 1

- 114 • *Burial (B)*: the pipeline is buried underneath the seabed and thus protected.
- 115 • *Exposure (E)*: the pipeline is exposed; visible and prone to damage. When the pipeline is exposed
 116 other pipeline features/events become visible:
 - 117 – *Anode (A)*: pipeline bracelet anodes are specifically designed to protect sub-sea pipelines
 118 from corrosion [18]. Data Coordinators visually recognise Anodes by the banding that
 119 appears in the orthogonal direction of the pipeline; anodes have no surface vegetation
 120 growth.
 - 121 – *Field Joint (FJ)*: the point where two pipe sections meet and welded together, typically
 122 occurring every 12 metres. Data Coordinators recognise Field Joints due to the depression
 123 on the pipeline surface.
 - 124 – *Free Span (FS)*: pipeline segments that are elevated and not supported by the seabed
 125 (either due to seabed erosion/scouring or due to uneven seabed during installation), pose
 126 significant risk to the asset; currents or moving objects (debris, nets and etc.) could damage

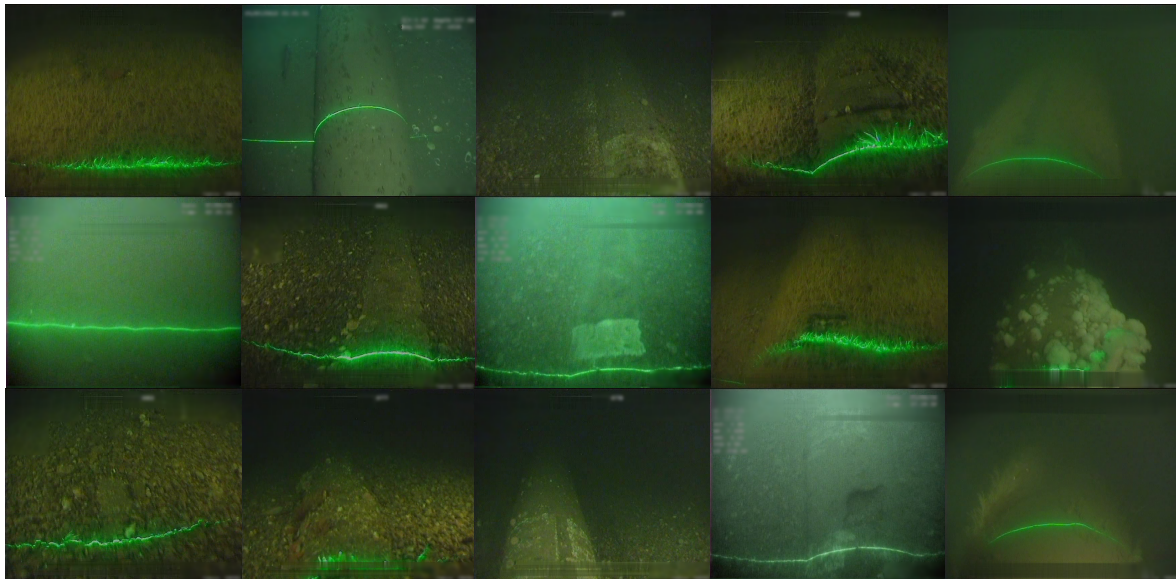


Figure 1. Examples of events in subsea pipeline surveys with varying scene conditions; from left to right: Burial, Exposure, Anode, Field Joint, Free Span

127 the pipeline. FS are more apparent on the starboard and port video feeds; the centre camera,
128 is used to judge the seabed depth against the pipeline.

129 The data set contains event (truthing) annotations created by trained Data Coordinators. It is
130 important to note that consecutive frames are highly correlated with each other and for that reason
131 still frames were extracted every 10 frames. The frames were labelled using a multi-label annotation
132 approach since events recorded during the pipeline survey are not mutually exclusive. The pipelines
133 are either buried underneath the seabed or exposed and thus visible. However, additional events
134 such as field joints, anodes and free spans are only observable when the pipeline is exposed. This
135 multi-label annotation approach is common practise in the scene classification domain, where images
136 may belong to multiple semantic classes [19]. The label distribution of the extracted frames is shown
137 in Figure 2. The data set contains 23,570 frames in total, consisting of 5,985 frames of burial, 4,236
138 frames of exposure, 6,119 frames of exposure and field joint, 2,494 frames of exposure and anode and
139 4,736 frames of exposure and free span. **Note, that all the annotated data (frames and labels) have been**
140 **checked for annotation correctness three times; one from the Data Coordinator on the vessel during**
141 **the execution of the survey, subsequently on-shore by the Quality Control (QC) personnel, and finally,**
142 **after the frames are extracted, by a trained Data Coordinator who confirmed the annotations through**
143 **manual inspection.**

R2.2

144 The first annotation procedure is performed by trained Data Coordinators on the vessel while the
145 data are captured. For the events, Exposure, Burial and Free Span, annotators do not solely rely on
146 video footage, but have information from the Multi-beam Echo which maps the seabed terrain. This
147 make annotation for these events consistent. The Anode and Field Joint events can be indeed missed
148 during the real-time annotation (although this is unlikely considering the training), this is quality
149 checked on shore (Step 2 below). The annotations are verified by a QC Data Coordinators in the office
150 before generating the client report. Routinely, QC Data Coordinators, have to their disposal annotation
151 data from previous surveys and “as built” information to corroborate the new survey. This eliminates
152 any missed events, especially the Anode and Field Joint events. Finally, when the frames extracted
153 from the survey data for training and testing dataset, we have performed further manual inspection to
154 ensure any inconsistencies of the labels are corrected..

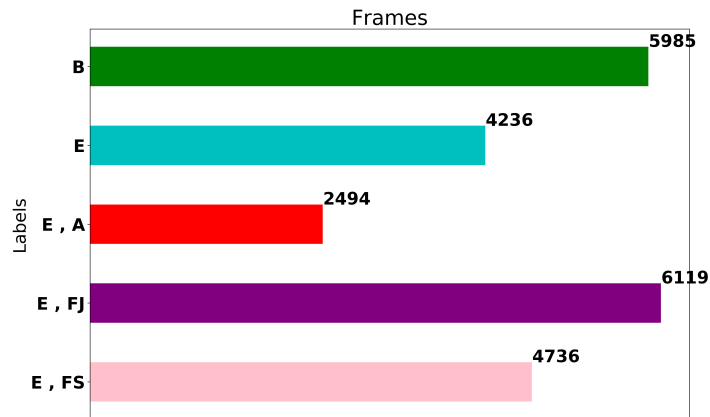


Figure 2. Label distribution of a total 23,570 frames of the complete dataset

155 2.1. Model Architecture

156 A Convolutional Neural Network (CNN) consists of three main types of layers: convolutional,
 157 pooling and fully connected. The convolutional layer consists of a set of independent filters which are
 158 individually convolved with the input image to generate a series of feature maps as an output [20].
 159 These filters can be adjusted to capture different features of interest within the image. The CNN
 160 utilised in the study is based on the ResNet architecture [21], the winner of the ImageNet Large Scale
 161 Visual Recognition Challenge 2015 [22]. ResNet is a state-of-the-art architecture that provides enhanced
 162 feature extraction capabilities for a wide range of applications, including being a backbone network
 163 for implementation of U-Net [23], RetinaNet [24], Faster R-CNN [25] and Mask R-CNN [26]. In this
 164 work, the ResNet-50 architecture is used that contains 25.6 M parameters. Other ResNet depths were
 165 examined to investigate the trade-off between performance and inference time (Section 6). Typically, a
 166 network with high number of parameters and network depth demands a large training data set to yield
 167 acceptable generalisation and performance. Creating a training data set of that size is expensive and
 168 laborious. An alternative approach is to adopt a transfer learning methodology, where a pre-trained
 169 network from a different domain is re-trained on data from the domain of interest (sub-sea pipeline
 170 inspection imagery in the present application). The pre-trained ResNet-50 network used is provided
 171 by PyTorch [27] trained on the ImageNet data set [22] comprising 1,000 image classes.

172 The ResNet-50 architecture, shown in Figure 3, consists of 5 stages; each stage comprising multiple
 173 layers of convolutions, Batch Normalisation [28] and Rectified Linear Unit (ReLU) activations [29]
 174 that do not affect the receptive fields of the convolutional layers [29]. More importantly, the ResNet
 175 architecture utilises the concept of skip (or identity) connections between stacked convolutional
 176 layers. These shortcut connections mitigate against the vanishing gradient problem on training deep
 177 architectures by allowing the gradients to propagate through identity connections. Maintaining
 178 the Feature Extraction layers is a standard methodology for application of transfer learning. In this
 179 case all the layers in the feature extractor are kept identical with the exception of the final pooling
 180 layer. After the fifth stage, the final layer consists of average and max pooling and then features are
 181 flattened and concatenated before being fed to two fully connected (linear) layers, with the purpose to
 182 reduce the dimensionality of the features and make the dimensions equal to the number of output
 183 labels. Furthermore, Batch Normalisation and Dropout layers are introduced between the linear
 184 layers to regularise the Head/Classifier. Note that the last linear layer for the pre-trained network
 185 consists of 1,000 output neurons to match the number of classes in the ImageNet data set; however
 186 in this application the output labels are 5 (Burial, Exposure, Free Span, Field Joint, Anode) and
 187 consequently the last layer is replaced by a linear layer containing 5 output neurons. The final layer
 188 is a Sigmoid activation function to squash network outputs between 0 and 1 independently for each
 189 neuron/label [30] using the equation:

R3.3 & R3.5

R1.2

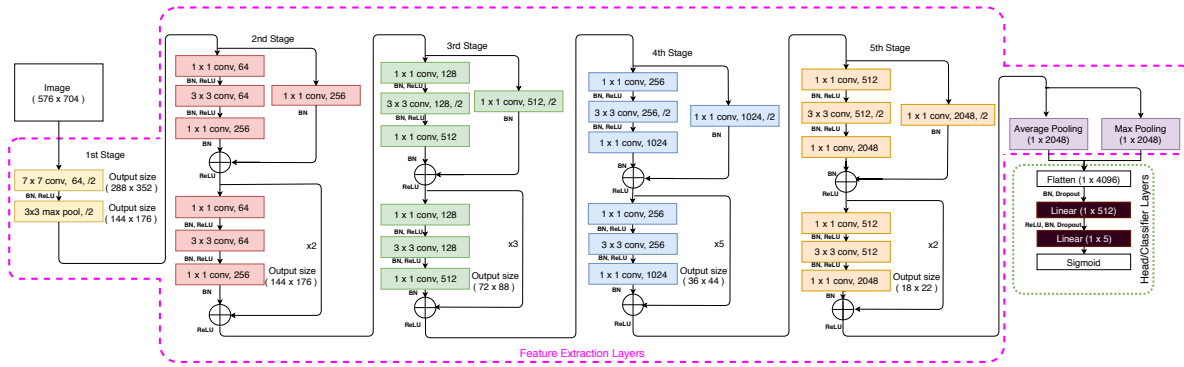


Figure 3. ResNet-50 Architecture with modified head.

$$\hat{y} = \sigma(z) = \frac{1}{1 + e^{-z}} \quad (1)$$

190 where z is the outputs of the last linear layer. The outputs of the network \hat{y} for an image would
 191 therefore be a vector of 5 real-valued numbers in the range 0 to 1 (one for each label) which can then
 192 be used to compute the sum of Binary Cross Entropy Loss for all labels:

$$\mathcal{L}(\hat{y}, y) = - \sum_{i=1}^C [y_i \cdot \log(\hat{y}_i) + (1 - y_i) \cdot \log(1 - \hat{y}_i)] \quad (2)$$

193 where C is number of labels, y is the one-hot encoded target (1 when the label is present in the ground
 194 truth data and 0 otherwise) and y_i is the element of that vector at location i . Similarly, \hat{y} is the predicted
 195 vector output of the network and \hat{y}_i is the element of the vector at location i which indicates the
 196 confidence level for the corresponding label.

197 2.2. Performance Evaluation Methodology

198 The Training, Validation and Testing methodology for the evaluation of the performance of the
 199 proposed network is shown in Figure 4. The full data set contains 23,570 frames with annotation
 200 according to the label distribution shown in Figure 2. Initially, 20% of the frames in the data set, are
 201 selected in a stratified fashion and set aside to be used as a test (keep-out) set and in the evaluation of
 202 the performance of the model after training/validation and hyper-parameter tuning. The methodology
 203 yields a test set of 4,714 frames with label distribution approximately equal to that shown in Figure 2.
 204 The remaining 80% (18,856 frames) of the data set is used to perform Monte Carlo Cross-validation [31]
 205 with stratified splits of 80/20% i.e. 80% of the data (15,085 frames) is used to train the model and its
 206 performance is validated on the remainder 20%; validation set (3,771 frames). The process is repeated
 207 multiple times (5 in this study) to evaluate the variability of the trained models and their performance
 208 on the validation sets. After hyper-parameter selection and tuning, the performance of the model is
 209 obtained on the test set to ensure representative performance on unseen data.

210 3. Model Training

211 In practice, training a deep CNN with random initialization for all its weights and biases is
 212 challenging, requiring a large data set given the large number of parameters that need to be adjusted.
 213 Consequently a common approach has been adopted, utilising Transfer Learning [32]. A neural
 214 network pre-trained on a large data set of images is used as a starting point. The rationale is that the
 215 initial layers of the pre-trained CNN are able to extract features that are generic for image classification
 216 tasks; e.g. edge detectors or color blob detectors. In the subsequent layers, network weights need to be
 217 fine-tuned to adapt to the specific features of the data set under consideration. In the present study, a
 218 deep CNN ResNet-50 [21] pre-trained on the ImageNet data set [33] is implemented (see Figure 3).

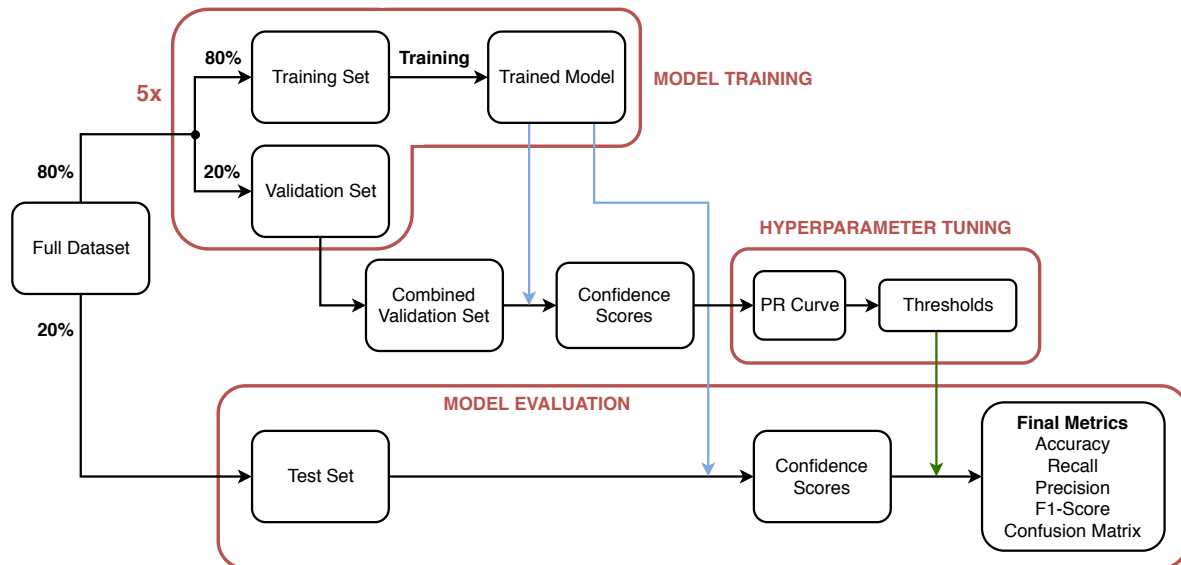


Figure 4. Model training and evaluation process.

219 The network can be logically divided in two sections; the feature extraction layers (enclosed in
 220 purple dashed lines in Figure 3) and the head or classification layers (enclosed in green dashed line
 221 in Figure 3). The weights of the feature extraction layers are initialised with the weights obtained
 222 from the pre-trained ResNet-50 network distributed with PyTorch 1.2.0 [34], while the head layers
 223 are randomly initialised. The Adam optimiser [35,36] is used for training with a mini batch size of 8
 224 and exponential decay parameters β_1 and β_2 equal to 0.9 and 0.99, respectively. Initially, when the
 225 head layers contain random weights, the loss function will yield high errors and thus there is a risk of
 226 disturbing the weights of the feature extraction layers when back-propagation is performed. For that
 227 reason, a multi-stage training approach is adopted.

228 In the first stage, training is performed for 4 epochs for only the weights of the last two fully
 229 connected layers of the network (shaded in Black in Figure 3), while the weights for all the other layers
 230 are frozen; i.e. weights are not updated. Furthermore, cyclic learning rate training [37] is utilised
 231 with maximum learning rate of 10^{-3} . The cyclic learning rate permits fast convergence and avoids
 232 local minima [38] during training. Subsequently, all the layers in the neural network are unfrozen
 233 and the network is trained for an additional 2 epochs. For these later epochs, cyclic learning rate is
 234 also adopted, however, different maximum learning rates for the Feature Extraction layers and the
 235 head are used; the maximum learning rates are set to 10^{-6} and 10^{-4} , respectively. A lower maximum
 236 learning rate is used for the feature extraction layers as their parameters are already well adjusted to
 237 extract generic image features. In contrast, the parameters of the head layers need more aggressive
 238 adjustment to fit the data set-specific features. Training is performed on a server equipped with two
 239 Nvidia GeForce RTX 2080 Ti, twelve Intel(R) Core(TM) i9-7960X CPU @ 2.80GHz and 128GB RAM.

240 Given the high capacity of the network, the risk of over-fitting of the training set needs to be
 241 evaluated. Two measures are taken to prevent over-fitting: regularisation through weight decay
 242 and online data augmentation. For weight decay, the regularisation parameter λ is set to 0.01 for all
 243 layers. Online data augmentation is used to increase the variability of the data set and enhance the
 244 generalisation of the model by limiting over-fitting [39]. A series of transformations are randomly
 245 applied to the training data, on every epoch, with probability of 75%, including horizontal flipping,
 246 rotation (with maximum angle of 10 degrees), scaling (with maximum variation of 1.05) and lighting
 247 alteration (with maximum variation change of 0.1). Data augmentation renders the model more robust
 248 and adaptable to the artifacts created, for example, by the motion of the ROV during the survey.

249 After training, the neural network outputs provide the confidence score for each label. Figure 5
 250 illustrates the confidence scores for each label, for the five selected events; the ground truth labels

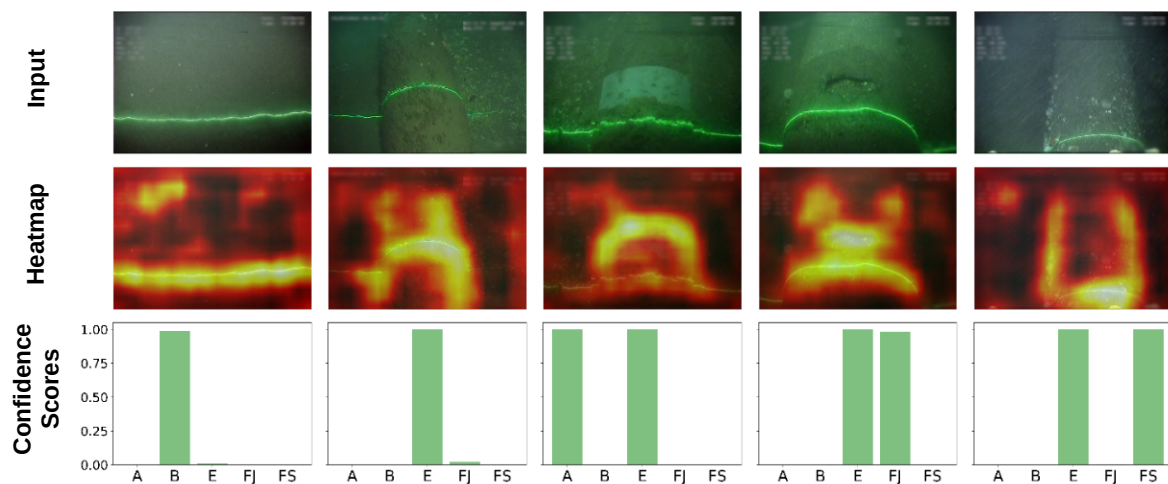


Figure 5. Ground Truth Label, Image, Heatmap and Predicted Confidence Scores for the five different event types

251 are shown at the top of each image. In all cases, the trained classifier yields high confidence scores
 252 (bottom bar chart) for these classes. The straight laser line observed when the pipeline is buried is
 253 the most relevant feature of the burial class, judging by the corresponding heat map (image on the
 254 middle row). When the pipeline is exposed, the model tends to focus on both its cylindrical shape
 255 and the curved nature of the laser line. In cases where other pipeline elements are visible, the model
 256 uses additional features to correctly classify the image. For example, for field joints, the unstructured
 257 depression/hole in the middle of the pipeline becomes a relevant feature; for anodes, the dominant
 258 feature is the characteristic white bracelet; for free spans, the most important feature is the well-defined
 259 edges of the pipeline resulting from its elevation with respect to the seabed.

260 The examples presented in Figure 5 have been intentionally extracted from the two different
 261 surveys and at different positions within each survey to highlight the large variation of image scenes.
 262 On consideration of the entire data set, these variations include differences in color (green, brown,
 263 grey), type of seabed (sand or gravels), vegetation (low or high) and distance and orientation of the
 264 ROV with respect to the seabed. The more variety the training set contains, the better the generalisation
 265 of the trained classifier will be.

266 4. Hyperparameter Tuning and Model Validation

267 After training, when an image is presented to the network input, the network output, after the
 268 final Sigmoid activation layer, is a vector with the degrees of confidence on whether or not each label
 269 is associated with the input image. In order to perform the final prediction and decide whether or not
 270 each label is associated with the input image, a threshold must be defined to make the output discrete; 1
 271 if confidence score exceeds threshold, 0 otherwise. The threshold can be either defined using a common
 272 value for all labels or defining five thresholds, one for each class/label [40,41]. Here, five separate
 273 thresholds are defined, one for each label to permit optimal performance per class. The selection of
 274 the thresholds is a means to adjust the sensitivity of the model for each label. Low thresholds will
 275 lead to high detection sensitivity at the expense of False Positives (FP), while high thresholds will
 276 reduce FPs at the expense of missed Positives [42]. The five threshold values constitute the model
 277 hyper-parameters and Precision/Recall Curves are used to determine optimal values, as illustrated
 278 in Figure 4. Precision-Recall curves are used in binary, and thus multi-label, classification to define
 279 a cut-off point (threshold) on the output confidence that the classifier assigns to each label and is
 280 commonly used in unbalanced data sets [43]. Note, that, the definition of the optimal thresholds is
 281 executed using solely the validation set, only containing images unseen during the training phase.

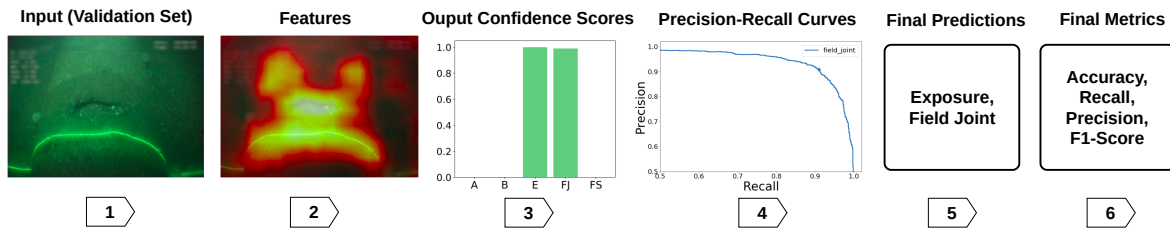


Figure 6. Steps for evaluating model's performance: (1) Validation Set (2) Feature Extraction (3) Classifier (4) Precision-Recall Curves for optimal thresholds selection (5) Applying optimal thresholds (6) Comparison with Ground Truth.

282 The evaluation of performance in multi-label learning is more challenging than in traditional
 283 single class settings, because each event can be associated with multiple labels simultaneously. In
 284 particular the following metrics are of interest:

$$Accuracy = \frac{TP + TN}{TP + FP + TN + FN} \quad (3)$$

$$Recall = \frac{TP}{TP + FN} \quad (4)$$

$$Precision = \frac{TP}{TP + FP} \quad (5)$$

$$F1-Score = \frac{2 \cdot Precision \cdot Recall}{Precision + Recall} \quad (6)$$

285 In this application, when metrics for a specific label are reported, the problem is reduced to a
 286 binary classification One-vs-Rest [44]. For instances where aggregate performance is reported, then the
 287 "micro" average [45] is computed. The exception is for aggregate accuracy, in which case, successful
 288 classifications counts are used only after all the labels have been identified correctly, commonly also
 289 known as "Exact Match Ratio (EMR)", a stricter metric, compared to average accuracy. Formally, the
 290 EMR is defined as:

$$ExactMatchRatio = \frac{1}{n} \sum_{i=1}^n \mathbb{1}(y_i = \hat{y}_i) \quad (7)$$

291 where $\mathbb{1}(y_i = \hat{y}_i)$ is the indicator function equal to 1 only when every element in the vector y_i is equal
 292 to every element in \hat{y}_i and n is the number of input samples. Note that for a binary classification (i.e.
 293 individual labels), this reduces to Accuracy.

294 Steps 1-4 in Figure 6 illustrate the process followed to obtain optimal threshold selection on
 295 the validation set. Note that due to 5-fold Monte Carlo cross-validation, five different models are
 296 trained, one for each validation fold. The predictions obtained from the five independent models
 297 on the five different validation folds are concatenated and used to determine the optimum set
 298 of thresholds/hyper-parameters. Precision-Recall curves can then be generated to evaluate the
 299 performance of the classifier at increasing values of confidence score thresholds. For each threshold
 300 value, the final set of predictions is evaluated against the corresponding ground truths at the individual
 301 label basis to identify each prediction as True Positive (TP), False Positive (FP), True Negative (TN) or
 302 False Negative (FN). The precision and recall of the classifier are then calculated using Equations 4
 303 and 5 (Step 4 in Figure 6). The optimum threshold is defined as the point that achieves the best balance
 304 between precision and recall, and therefore corresponds to the closest point to the top right corner
 305 on the graph (coordinate (1,1)). The strategy to define the optimal threshold was selected because in this
 306 application, it is equally important to maximise precision and recall to provide the maximum F1-score.

307 Applying the methodology for the five event types (Anode, Burial, Exposure, Field Joint and
 308 Free Span), results in the Precision-Recall curves shown in Figure 7. The optimal thresholds are at

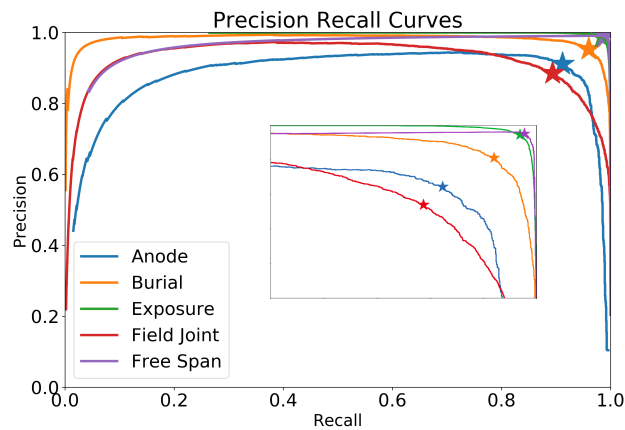


Figure 7. Precision-Recall curves for all labels. The inset shows a zoomed version of the top right corner.

Event	Anode	Burial	Exposure	Field Joint	Free Span
Threshold	0.357	0.367	0.632	0.542	0.430

Table 1. Optimum label-based thresholds for the validation set.

Fold #	Exact Match Ratio	Precision	Recall	F1-Score
1	0.907	0.958	0.961	0.960
2	0.890	0.949	0.956	0.953
3	0.920	0.972	0.961	0.967
4	0.914	0.962	0.967	0.964
5	0.899	0.954	0.958	0.956

Table 2. Aggregate performance of the five models, one for each fold.

309 the locations depicted by the star (“*”) carets in the graph and yield thresholds for each event type,
 310 summarised in Table 1.

311 Using the optimal thresholds identified from hyper-parameter tuning, the performance metrics
 312 (Equations 3-7) for each model in their corresponding validation fold is shown in Table 2.

313 Similarly, the average performance of the five models for each event type is shown in Table 3
 314 along with the standard deviation for each metric. Field Joints are the most challenging class with the
 315 lowest F1-score of 88.9%, expected given that such events are often difficult to distinguish due to the
 316 subtle features. On the other extreme, free spans and exposures show high performance, with F1-score
 317 of 98.8% and 98.5%, respectively. The aggregate F1-score (micro-average) is 96%.

Event	Threshold	Accuracy		Recall		Precision		F1-Score	
		Average	Std	Average	Std	Average	Std	Average	Std
Anode	0.357	0.981	0.006	0.910	0.028	0.912	0.046	0.911	0.028
Burial	0.367	0.978	0.001	0.959	0.011	0.953	0.013	0.956	0.004
Exposure	0.632	0.978	0.001	0.984	0.004	0.986	0.003	0.985	0.001
Field Joint	0.542	0.942	0.008	0.893	0.020	0.885	0.024	0.889	0.015
Free Span	0.430	0.995	0.002	0.988	0.002	0.988	0.013	0.988	0.007
Aggregate		0.906	0.011	0.961	0.004	0.959	0.008	0.960	0.005

Table 3. Metrics with optimal thresholds on the validation set.

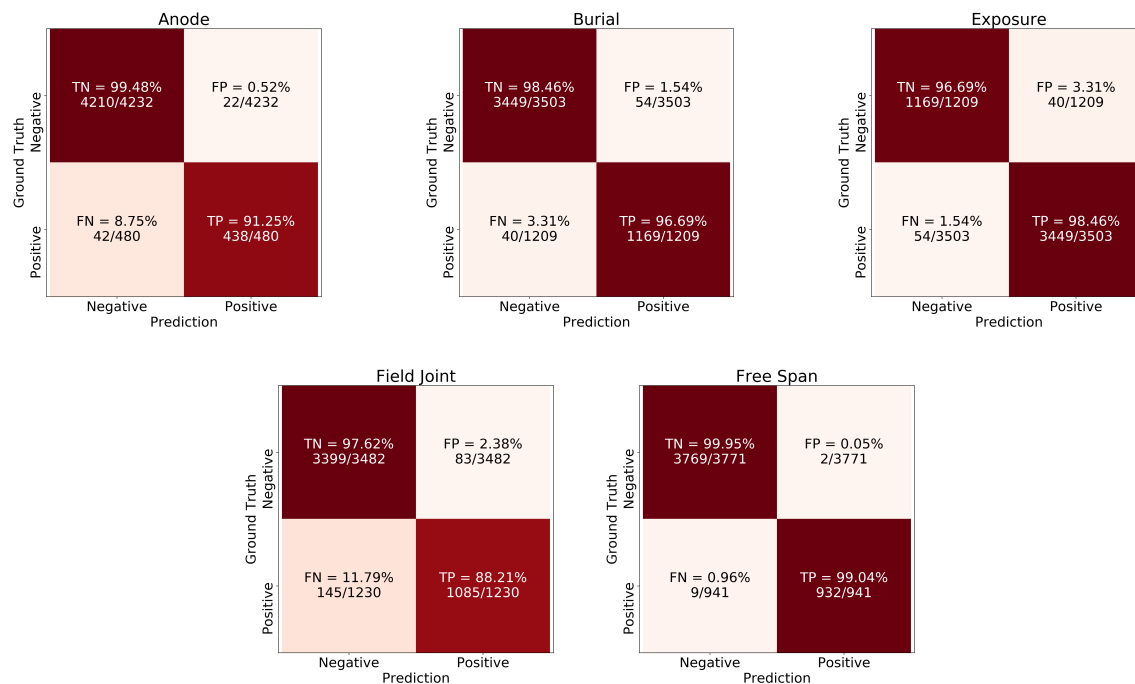


Figure 8. Confusion matrices on the test set for each class; Anode, Burial, Exposure, Field Joint and Free Span.

5. Model Performance on Test Set

In order to ensure that thresholds are not biased to the validation set, the final model performance evaluation is carried out on a previously unseen (keep-out) test set (Figure 4); i.e. the images that have not been used for either training nor validation or hyper-parameter tuning. The cross-validation has yielded five different models and the model selected for final testing is the one that provides the highest F1-Score viz. the model of the third fold, shown in bold in Table 2. Figure 8 shows the confusion matrices for each label, obtained using the final model on the test set. Each label is considered positive if it is present in the image frame and negative otherwise. The confusion matrices show the absolute number of frames and the percentage of TN, FP, FN, TP. For instance, the total number of frames in the test set is 4,712 frames with 480 frames associated with the label “Anode” and 4,232 are not. From the 480 frames that are labelled as “Anode” (positive frames), 438 (91.25%) have been correctly identified by the model (TP) and 42 (8.75%) have been missed (FN). In terms of FP, 22 frames have been incorrectly identified as anodes out of 4,232 i.e. a False Positive Rate of 0.52%.

From the confusion matrices, the Field Joints are the most challenging label with miss rate of 11.79% and false positive rate of 2.38%. FJ mis-classifications can be attributed to visual artefacts in the imagery, for example small rocks or vegetation. It is worth noting that this is the classifier performance on a single frame basis; when the classifier is applied on a video stream with 25 fps, the probability that these artefacts appearing in all frames is reduced and as a consequence the probabilities of a missing event or incorrect identification reduces, respectively. The performance of the network on per label basis is summarised in Table 4. Overall, the Accuracy (Exact Match Ratio) of the network is 91.9% with F1-Score of 96.6%.

6. Effect of Model Size

Identical evaluation performance was carried out for ResNet models with 18, 34, 101 and 152 layers (in addition to 50). The resultant performance metrics, on the test set, for each model size are summarised in Table 5. As the model complexity and capacity increase, the F1-Scores initially improves until the ResNet-50 architecture. Further increases in the number of layers (i.e. 101 and 152), result in

Event	Threshold	Accuracy	Precision	Recall	F1-Score
Anode	0.357	0.986	0.952	0.912	0.931
Burial	0.367	0.980	0.955	0.966	0.961
Exposure	0.632	0.980	0.988	0.984	0.986
Field Joint	0.542	0.951	0.928	0.882	0.904
Free Span	0.430	0.997	0.997	0.990	0.994
Aggregate		0.919	0.972	0.960	0.966

Table 4. Test set performance of individual labels and aggregate.

Network	# Parameters	Inference Time (ms)	Exact Match Ratio	Precision	Recall	F1-Score
ResNet-18	11,706,949	17.7	0.872	0.945	0.947	0.946
ResNet-34	21,815,109	20.8	0.903	0.953	0.966	0.960
ResNet-50	25,617,477	23.6	0.919	0.972	0.960	0.966
ResNet-101	44,609,605	31.2	0.916	0.956	0.973	0.965
ResNet-152	60,253,253	39.1	0.833	0.931	0.927	0.929

Table 5. Test set performance of different ResNet model sizes.

344 performance degradation. Larger models have the tendency to overfit faster. This is likely to occur
 345 given the training parameters are kept identical; i.e. number of epochs, regularisation coefficients,
 346 learning rates and etc. and altering these parameters may be necessary to achieve optimal prediction
 347 accuracy. Even though larger networks have the potential to achieve better F1-Score, as the number
 348 of layers increase, the number of parameters increase significantly along with the inference times.
 349 Note that inference time reports in Table 5 is average computation time over 100 frame predictions;
 350 i.e. 100 forward passes. For the deeper networks the inference time are marginally within the bounds
 351 of real-time operation. From these results, the ResNet-50 model is selected as it provides the best
 352 performance with inference time within the bounds of real-time operation.

353 7. Conclusions

354 A ResNet-50 deep convolutional neural network is employed to automatically detect and annotate
 355 five sub-sea survey events; Anode, Exposure, Burial, Field Joint and Free Span relying exclusively
 356 on the centre video feed of an ROV. To minimise the challenging demands on the scope of the
 357 training data, a transfer learning approach is adopted where the feature extraction layers of the
 358 network are initialised using the weights of a network pre-trained on ImageNet. The head of the
 359 network is adjusted to permit multi-label classification yielding the identification of events appearing
 360 concurrently in the video frames. Subsequently, the developed network is re-trained on 23,570 images
 361 extracted from real survey data. Several network depths were tested and the ResNet-50 network
 362 was selected to balance the trade-off between performance and inference time. The network has
 363 been evaluated on a test keep-out set to measure its ability to generalise. The framework achieves
 364 an Exact Match Ratio (i.e. all labels identified correctly) of 91.9% and a F1-Score ‘micro’-average of
 365 96.6%. The most challenging class to detect are Field Joints which have been detected with accuracy of
 366 95.1% and F1-Score of 90.4%, respectively. The metrics are obtained on a single-frame basis and the
 367 proposed network is able to classify frames within 23.6 ms on an NVIDIA GeForce RTX 2080 Ti GPU,
 368 effectively executing real-time classification of video streams at 25 fps. Results along with the real-time
 369 operation of the network demonstrate that automatic video annotation has the potential to increase
 370 the speed of survey execution, increase the consistency of annotation and reduce the demand on
 371 off-shore personnel, benefiting health and increasing safety. Future work will investigate the benefits
 372 in combining predictions from consecutive frames and the fusion of the video data with multi-beam
 373 echo, pipe-tracker instrumentation to improve annotation performance.

R3.3 & R3.5

374 **Acknowledgments:** The work was partially supported by The Data Lab Innovation Centre, Edinburgh, Scotland,
375 UK (project registration code 16270), the Oil and Gas Innovation Centre, Aberdeen, Scotland UK (project
376 registration code 18PR-16) and N-Sea, Zierikzee, Netherlands. The Data Lab and the Oil and Gas Innovation
377 Centres are funded by the Scottish Funding Council through the Innovation Centres Programme.

378 References

- 379 1. Zingaretti, P.; Zanolli, S.M. Robust real-time detection of an underwater pipeline. *Engineering Applications*
380 *of Artificial Intelligence* **1998**, *11*, 257–268. doi:10.1016/S0952-1976(97)00001-8.
- 381 2. Jacobi, M.; Karimanzira, D. Underwater pipeline and cable inspection using autonomous underwater
382 vehicles. 2013 MTS/IEEE OCEANS - Bergen, 2013, pp. 1–6. doi:10.1109/OCEANS-Bergen.2013.6608089.
- 383 3. Jacobi, M.; Karimanzira, D. Multi sensor underwater pipeline tracking with AUVs. 2014 Oceans - St.
384 John's; IEEE: St. John's, NL, 2014; pp. 1–6. doi:10.1109/OCEANS.2014.7003013.
- 385 4. Narimani, M.; Nazem, S.; Loueipour, M. Robotics vision-based system for an underwater pipeline and
386 cable tracker. OCEANS 2009-EUROPE, 2009, pp. 1–6. doi:10.1109/OCEANSE.2009.5278327.
- 387 5. Conte, G.; Zanolli, S.; Perdon, A.M.; Tascini, G.; Zingaretti, P. Automatic analysis of visual data in submarine
388 pipeline inspection. OCEANS 96 MTS/IEEE Conference Proceedings. The Coastal Ocean - Prospects for
389 the 21st Century, 1996, Vol. 3, pp. 1213–1219 vol.3. doi:10.1109/OCEANS.1996.569075.
- 390 6. Ortiz, A.; Simó, M.; Oliver, G. A vision system for an underwater cable tracker. *Machine Vision and*
391 *Applications* **2002**, *13*, 129–140. doi:10.1007/s001380100065.
- 392 7. Ortiz, A.; Antich, J.; Oliver, G. Experimental Evaluation of a Particle Filter-based Approach for Visually
393 Tracking Undersea Cables. *IFAC Proceedings Volumes* **2009**, *42*, 140 – 145. 8th IFAC Conference on
394 Manoeuvring and Control of Marine Craft, doi:10.3182/20090916-3-BR-3001.0041.
- 395 8. Asif, M.; Rizal, M. An Active Contour and Kalman Filter for Underwater Target Tracking and Navigation.
396 In *Mobile Robots: towards New Applications*; Lazinica, A., Ed.; I-Tech Education and Publishing, 2006.
397 doi:10.5772/4699.
- 398 9. Nguyen, V.N.; Jenssen, R.; Roverso, D. Automatic autonomous vision-based power line inspection: A
399 review of current status and the potential role of deep learning. *International Journal of Electrical Power &*
400 *Energy Systems* **2018**, *99*, 107 – 120. doi:10.1016/j.ijepes.2017.12.016.
- 401 10. Zhang, W.; Witharana, C.; Li, W.; Zhang, C.; Li, X.; Parent, J. Using deep learning to identify utility poles
402 with crossarms and estimate their locations from google street view images. *Sensors (Switzerland)* **2018**.
403 doi:10.3390/s18082484.
- 404 11. Jalil, B.; Leone, G.R.; Martinelli, M.; Moroni, D.; Pascali, M.A.; Berton, A. Fault Detection in Power
405 Equipment via an Unmanned Aerial System Using Multi Modal Data. *Sensors* **2019**. doi:10.3390/s19133014.
- 406 12. Miao, X.; Liu, X.; Chen, J.; Zhuang, S.; Fan, J.; Jiang, H. Insulator detection in aerial images for transmission
407 line inspection using single shot multibox detector. *IEEE Access* **2019**. doi:10.1109/ACCESS.2019.2891123.
- 408 13. Ren, S.; He, K.; Girshick, R.; Sun, J. Faster R-CNN, 2017, [1506.01497]. doi:10.1109/TPAMI.2016.2577031.
- 409 14. Bonnin-Pascual, F.; Ortiz, A. A novel approach for defect detection on vessel structures using
410 saliency-related features. *Ocean Engineering* **2018**, *149*, 397 – 408. doi:10.1016/j.oceaneng.2017.08.024.
- 411 15. Bonin-Font, F.; Campos, M.M.; Codina, G.O. Towards Visual Detection, Mapping and Quantification
412 of *Posidonia Oceanica* using a Lightweight AUV. *IFAC-PapersOnLine* **2016**, *49*, 500 – 505. 10th IFAC
413 Conference on Control Applications in Marine SystemsCAMS 2016, doi:10.1016/j.ifacol.2016.10.485.
- 414 16. Martin-Abadal, M.; Guerrero-Font, E.; Bonin-Font, F.; Gonzalez-Cid, Y. Deep Semantic Segmentation
415 in an AUV for Online *Posidonia Oceanica* Meadows Identification. *IEEE Access* **2018**, *6*, 60956–60967.
416 doi:10.1109/ACCESS.2018.2875412.
- 417 17. Petraglia, F.R.; Campos, R.; Gomes, J.G.R.C.; Petraglia, M.R. Pipeline tracking and event classification
418 for an automatic inspection vision system. 2017 IEEE International Symposium on Circuits and Systems
419 (ISCAS); IEEE: Baltimore, MD, USA, 2017; pp. 1–4. doi:10.1109/ISCAS.2017.8050761.
- 420 18. Fang, H.; Duan, M. Submarine Pipelines and Pipeline Cable Engineering. In *Offshore Operation Facilities*;
421 Elsevier, 2014; pp. e1–e181. doi:10.1016/B978-0-12-396977-4.00006-8.
- 422 19. Boutell, M.R.; Luo, J.; Shen, X.; Brown, C.M. Learning multi-label scene classification. *Pattern Recognition*
423 **2004**, *37*, 1757–1771. doi:10.1016/j.patcog.2004.03.009.
- 424 20. Sinha, R.K.; Pandey, R.; Pattnaik, R. Deep Learning For Computer Vision Tasks: A review. *CoRR* **2018**,
425 *abs/1804.03928*, [1804.03928].

- 426 21. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep Residual Learning for Image Recognition. *arXiv:1512.03385 [cs]*
427 **2015**. arXiv: 1512.03385.
- 428 22. Russakovsky, O.; Deng, J.; Su, H.; Krause, J.; Satheesh, S.; Ma, S.; Huang, Z.; Karpathy, A.; Khosla, A.;
429 Bernstein, M.; Berg, A.C.; Fei-Fei, L. ImageNet Large Scale Visual Recognition Challenge. *International*
430 *Journal of Computer Vision (IJCV)* **2015**, *115*, 211–252. doi:10.1007/s11263-015-0816-y.
- 431 23. Ronneberger, O.; Fischer, P.; Brox, T. U-Net: Convolutional Networks for Biomedical Image Segmentation.
432 *arXiv:1505.04597 [cs]* **2015**. arXiv: 1505.04597.
- 433 24. Lin, T.Y.; Goyal, P.; Girshick, R.; He, K.; Dollár, P. Focal Loss for Dense Object Detection. *arXiv:1708.02002*
434 *[cs]* **2017**. arXiv: 1708.02002.
- 435 25. Ren, S.; He, K.; Girshick, R.; Sun, J. Faster R-CNN: Towards Real-Time Object Detection with Region
436 Proposal Networks. *arXiv:1506.01497 [cs]* **2015**. arXiv: 1506.01497.
- 437 26. He, K.; Gkioxari, G.; Dollár, P.; Girshick, R. Mask R-CNN. *arXiv:1703.06870 [cs]* **2017**. arXiv: 1703.06870.
- 438 27. Paszke, A.; Gross, S.; Massa, F.; Lerer, A.; Bradbury, J.; Chanan, G.; Killeen, T.; Lin, Z.; Gimelshein, N.;
439 Antiga, L.; Desmaison, A.; Kopf, A.; Yang, E.; DeVito, Z.; Raison, M.; Tejani, A.; Chilamkurthy, S.; Steiner,
440 B.; Fang, L.; Bai, J.; Chintala, S. PyTorch: An Imperative Style, High-Performance Deep Learning Library.
441 In *Advances in Neural Information Processing Systems 32*; Curran Associates, Inc., 2019; pp. 8024–8035.
- 442 28. Ioffe, S.; Szegedy, C. Batch Normalization: Accelerating Deep Network Training by Reducing Internal
443 Covariate Shift. *arXiv:1502.03167 [cs]* **2015**. arXiv: 1502.03167.
- 444 29. Nair, V.; Hinton, G.E. Rectified Linear Units Improve Restricted Boltzmann Machines. Proceedings of the
445 27th International Conference on International Conference on Machine Learning; Omnipress: USA, 2010;
446 ICML'10, pp. 807–814.
- 447 30. Nwankpa, C.; Ijomah, W.; Gachagan, A.; Marshall, S. Activation Functions: Comparison of trends in
448 Practice and Research for Deep Learning. *arXiv:1811.03378 [cs]* **2018**. arXiv: 1811.03378.
- 449 31. Geisser, S. The Predictive Sample Reuse Method with Applications. *Journal of the American Statistical*
450 *Association* **1975**, *70*, 320–328.
- 451 32. Pan, S.J.; Yang, Q. A Survey on Transfer Learning. *IEEE Transactions on Knowledge and Data Engineering*
452 **2010**, *22*, 1345–1359. doi:10.1109/TKDE.2009.191.
- 453 33. Deng, J.; Dong, W.; Socher, R.; Li, L.J.; Kai Li.; Li Fei-Fei. ImageNet: A large-scale hierarchical image
454 database. 2009 IEEE Conference on Computer Vision and Pattern Recognition; IEEE: Miami, FL, 2009; pp.
455 248–255. doi:10.1109/CVPR.2009.5206848.
- 456 34. Paszke, A.; Gross, S.; Chintala, S.; Chanan, G.; Yang, E.; DeVito, Z.; Lin, Z.; Desmaison, A.; Antiga, L.; Lerer,
457 A. Automatic Differentiation in PyTorch. NIPS Autodiff Workshop, 2017.
- 458 35. Kingma, D.P.; Ba, J. Adam: A Method for Stochastic Optimization, 2014. cite arxiv:1412.6980Comment:
459 Published as a conference paper at the 3rd International Conference for Learning Representations, San
460 Diego, 2015.
- 461 36. Reddi, S.J.; Kale, S.; Kumar, S. On the Convergence of Adam and Beyond. International Conference on
462 Learning Representations, 2018.
- 463 37. Smith, L.N. Cyclical Learning Rates for Training Neural Networks. 2017 IEEE Winter Conference on
464 Applications of Computer Vision (WACV), 2017, pp. 464–472. doi:10.1109/WACV.2017.58.
- 465 38. Smith, L.N. Cyclical Learning Rates for Training Neural Networks. *arXiv:1506.01186 [cs]* **2015**. arXiv:
466 1506.01186.
- 467 39. Guo, Y.; Liu, Y.; Oerlemans, A.; Lao, S.; Wu, S.; Lew, M.S. Deep learning for visual understanding: A
468 review. *Neurocomputing* **2016**, *187*, 27–48. doi:10.1016/j.neucom.2015.09.116.
- 469 40. Sorower, M.S. A literature survey on algorithms for multi-label learning. Technical report, Oregon State
470 University, Corvallis, 2010.
- 471 41. Gharroudi, O.; Elghazel, H.; Aussem, A. Ensemble Multi-label Classification: A Comparative Study on
472 Threshold Selection and Voting Methods. 2015 IEEE 27th International Conference on Tools with Artificial
473 Intelligence (ICTAI); IEEE: Vietri sul Mare, Italy, 2015; pp. 377–384. doi:10.1109/ICTAI.2015.64.
- 474 42. Flach, P.A.; Kull, M. Precision-Recall-Gain Curves: PR Analysis Done Right. Proceedings of the 28th
475 International Conference on Neural Information Processing Systems - Volume 1; MIT Press: Cambridge,
476 MA, USA, 2015; NIPS'15, pp. 838–846.
- 477 43. Saito, T.; Rehmsmeier, M. The Precision-Recall Plot Is More Informative than the ROC Plot When Evaluating
478 Binary Classifiers on Imbalanced Datasets. *PLOS ONE* **2015**, *10*, 1–21. doi:10.1371/journal.pone.0118432.

- 479 44. Zhang, M.L.; Zhou, Z.H. A Review on Multi-Label Learning Algorithms. *IEEE Transactions on Knowledge*
480 *and Data Engineering* **2014**, *26*, 1819–1837. doi:10.1109/TKDE.2013.39.
- 481 45. Yang, Y. An Evaluation of Statistical Approaches to Text Categorization. *Inf. Retr.* **1999**, *1*, 69–90.
482 doi:10.1023/A:1009982220290.

483 © 2020 by the authors. Submitted to *Sensors* for possible open access publication under the terms and conditions
484 of the Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0/>).