BBRC
Bioscience Biotechnology
Research Communications

Intelligent Agent for Automatic Engineering Diagram Digitization with Deep Learning

Premanand Ghadekar[1] and Shaunak Joshi[2]
[1]Head & Associate Professor, Dept. of IT, VIT Pune, India
[2]Research Scholar, Dept. of IT, VIT Pune, India

ABSTRACT

Detailed diagrammatical representations used in the process industry used for representing process flow & piping equipment coupled alongside with instruments and control devices in the process industry are called Piping and Instrumentations Diagrams (P&ID). Up until recently, P&IDs were often handmade, scanned then stored as images and the digitizing of P&IDs used to be a slow time-consuming task, meticulously done by humans. Extracting P&ID data is a challenging task since there are multiple vision challenges associated with it such as the varied resolutions of different P&IDs and the presence of noise in the image. Considering all the issues, traditional computer vision has faced considerable roadblocks in P&ID data extraction. Using the recent advances in deep learning the proposed model proposes an end to end GPU based algorithm with modular subparts which would semantically analyze, extract P&ID data. The accuracy and correctness are of prime importance in the data extraction as it enables the project to be exact on its costing estimates which helps to maintain profitability. The proposed algorithm identifies each instance of valves, equipment's and instruments used in the P&ID and extracts its meta data as well. Then it identifies all the lines and the text present in the P&ID. In the final step all the P&ID components are associated to their respective meta data and link them to their respective pipe-codes which in turn are mapped to the pipelines which the components are hosted. As per the current viability no system does both these tasks end to end with high levels of accuracy.

KEY WORDS: P&ID, OBJECT DETECTION, DOCUMENT SEMANTICS, RETINA NETWORK, FOCAL LOSS, CONVOLUTIONAL NETWORK, GPU.

INTRODUCTION

Piping & Instrumentation Diagrams (P&ID) are a basic building block / integral part of any heavy engineering project. Each project has thousands of P&ID files which are collectively placed in a digital ledger. The accuracy and correctness are of these engineering diagrams are of prime importance as it helps the engineering companies building the project to be exact on their requirements. P&IDs are in use through the entire life cycle of a project beginning at the construction and all including operations phase & in order to convey material needs of the facility, these diagrams are created. In many countries, keeping the P&ID updated is legally required. Since process engineers use the P&ID to plan the industrial facility and analyze its components while using it also to keep a tab on any changes.

All these points make P&ID's an invaluable resource for data with respect to the industrial facility. However, there is little cohesion in these diagrams as they largely stored in archival format as CAD/CAM or PDF files with no metadata describing their components and connections between the different P&IDs. As intelligent sensors based on the IoT have seen increasing use in these industrial plants. The data they generate is often critical to the real

01

time operation of the plant. Sensor contextualization and linking them to external systems is an important part of the puzzle.

Figure 1: The proposed 3-step process for Information extraction from Piping and Instrumentation Diagrams
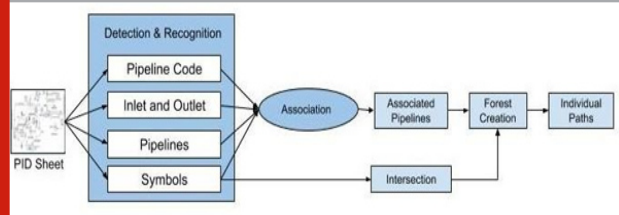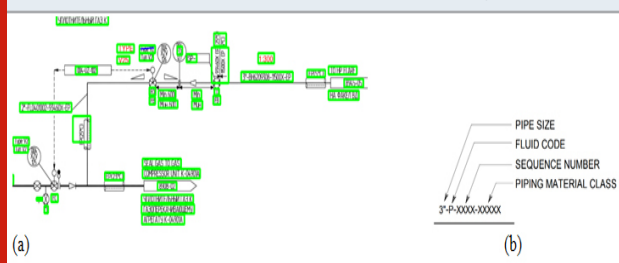

Figure 2: Text detection results from the FOTs based neural network model coupled with the pipecode format used, b. Code format which was used to make the Regex.

This entire step of manually digitizing the P&IDs is error prone and time consuming. This paper proposes exploiting cutting edge techniques in machine intelligence, computer vision and text recognition, to create a GPU based deep learning algorithmic pipeline which creates a hierarchical and structured representation which takes in account the interlinking of various components of the engineering diagram such as and not limited to equipment, pipe codes, text, etc. and give out a high fidelity output which would be component hierarchy. Which can be used for a multitude of tasks such as machine learning analytics, visualization and life cycle management.

Related Algorithms: Until very recently there was very little work done for semantically understanding Piping & Instrumentation Diagram (P&ID), however recently there has been a soft boom in the areas of object detection and classification in terms of digitization of complex engineering diagrams & document recognition. An automated method for converting DXF format to SVG (Adam, S et al., 2000) while preserving meta data. A case-based approach framework for engineering drawing recognition was proposed by (Arroyo, E. et al, 2014). In (Arroyo, E. et al., 2015) tried using vector-graphics-coded engineering documents to automate the extraction of association information. A partial matching method and its spatial relation graph (SRG) used in composite graphics recognition in (Belongie, S. et al., 2002).

Fourier Mellin Transforms were proposed by (Fei, L. et al., 2009) to classify multi-oriented and multi-scaled patterns. Auto Associative neural networks (Fei, L. et al. 2011), Deep Belief networks (Gellaboina, M. K. et al.

2009), and consistent attributed graphs (CAG) (Goh, K. N et al. 2013) were all tried for relation markers. Certain models that use visual feature maps which capture online stroke properties (Gupta, G. et al., 2017) and shape-based (Kiryati, N.et al., 1991). Connected Component (CC) analysis (Koo, H. I. et al.,2013) has been used extensively for extracting characters (Ouyang, T. Y. et al., 2009) from images. To detect text from the given media a CTP Network (Verma, A. et al., 2016) was used.

A Probabilistic Hough Transform (Saha, P. K., et al., 2016) variant was used to detect line segments, PHT is preferred over the standard version as its more robust and efficient while being quicker at inference due to the implementation using random sampling of edge points. In terms of pipelines, (Sung-O Kang et al., 2019) proposed a system, using template matching, sliding window approach and other simplistic methods such as to detect lines, text and symbols. Template matching though easily implemented has issues such as a failure to generalize well due to intra class variations across projects or a need for a large symbol database which captured every variation in every symbol. An innovative approached spearheaded by (Dries Van Daele et al., 2019)used image segmentation and object recognition techniques and tabular data reading techniques to represent the data in a feature vector to enable quick data extraction. For symbol detection and image segmentation traditional classification networks fall short due to which Retina Net (Ren, S. et al., 2015)is used as the symbols found in the P&IDs have a high occurrence of noise and large intraclass and minute inter-class differences coupled with the presence of text present inside the symbols.

Approach & Methodology: The proposed approach involves using Deep learning and Traditional Computer Vision Techniques in tandem coupled with computational algorithms to create a complete picture of the P&ID in highly structured and hierarchically layered data format. The aim of this project is the creation of an artificially intelligent agent which is used to achieve a high degree of accuracy in P&ID data extraction, faster workflow and to save costly man-hours of work. This intelligent agent is proposed as a novel aid to an engineer to make the highly tedious task of reading P&IDs and digitizing them a highly efficient process. The high-level idea of the proposed Algorithm is using GPU based Deep Learning models for Object Recognition & Text Semantics in a graphical compute rich environment coupled with core sub algorithms for semantic understanding and mapping of generated data present in the P&ID.

This is a multistage process in which information extraction from the P&ID sheet is done: (1) Initially, individual identification of all the components in the P&ID is done like the pipelines, pipeline codes, symbols, connectors, instruments, spec breaks & text. Conventional computer vision techniques and geometric algorithms were used here such as Point Detection Algorithms, Hough Transform, to detect and localized the geometric properties of the P&ID components, pipeline codes, deep neural networks-based Symbol detector is

used. (2) For the second step, these components which are separately identified, are mapped and stored in data structures which store the local temporal information about the P&ID component and its meta data. (3) Finally, composite random forest-based data structure is used to link all the components such as lines, pipe codes, symbols, equipment's and instruments by the direction of flow of the pipeline. This data structure represents the different relationships and mappings each component has to other components and give us the P&ID in a hierarchal, data format which is finally refined into a digitized P&ID. Figure 1. Shows the proposed 3step process for Information extraction from Piping and Instrumentation Diagrams.

Detection and Recognition

Pipeline Code Detection: Every pipeline code is unique and used to differentiate and distinguish each pipeline. Hence, the first step in this project is to identify the pipeline code. For images a FOTS (Xuebo Liu et al., 2018) network is used to extract the text. Since pdf is encoded in a .XML tree format, it is converted to a scalable vector graphics (SVG) format. P&I Diagram is plotted in 2D space and spatially encode the text boxes. Finally, geometric transforms are used to understand the orientation, this is done using orientation-based matrices which are multiplied with the co-ordinates of the text box to determine the orientation. This method returns all the possible candidate's components for text in the P&I Diagram sheet with 100% recall however a significant no of texts extracted which are unnecessary are removed later. Since it is known that each pipeline code length is fixed and is of a standard format, powerful regular expressions are used to weed out the spurious text detections. Text detection results from the FOTs based neural network model coupled with the pipecode format used is given in Figure 2.

2 Inlet & Outlet Detection: Any pipelines end or starts are decided by inlets and outlets. An inlet and outlet are made up of 5 vertices the width of 3 times with respect to its height. These exact physical properties were used here to detect the inlet/outlet in a robust manner using some heuristics. For detection of the inlets, first extract the text blobs detected in the previous iteration for further processing. Douglas Ramer algorithm was utilized to identify the polygons. Which are then further used to classify into inlets and outlets. There can only be 4 types of polygons. Using Douglas-Ramer Algorithm it is seen that there are two sides to an inlet/outlet each having 3 or 2 points. Since an inlet or an outlet can be easily deciphered if there is pipeline passing through the side having 3 points and if the side with 2 points is at a sufficient distance with respect to the other side, this test can be used vice versa and finally it is checked by application of a small mask on both sides and verifying if a line passes through it. Figure 3 shows Inlets and outlets in P & D.

3. Detection of Pipeline: Here all the lines from the .XML DOM are extracted from the image, text and the inlets and the outlets which are previously extracted from the

images, are removed. The lines are then plotted according to the co-ordinates found in the Scalable Vector Graphics Data. These points are then scaled accordingly and then OTSU thresholding is used to clean up the spurious lines.

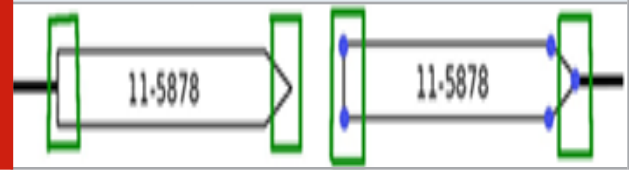Figure 3: Inlets and outlets in P&I Diagrams

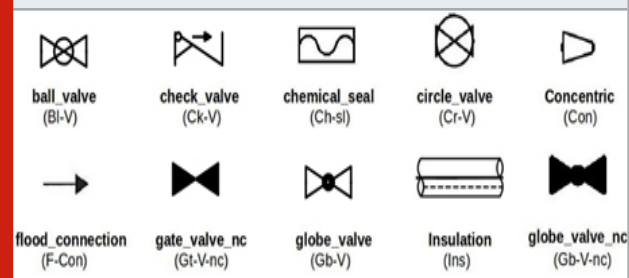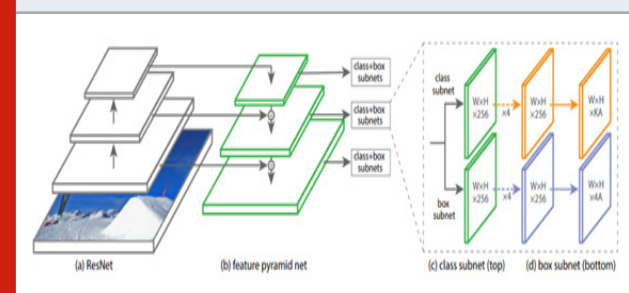Figure 4: P&ID Sheet symbols which are used as classes

Figure 5: Retina-Net Architecture: (1) Base: Stacked feature network based on ResNet101. (2) Class subnet: finds if the object is present or not. (3) Box Subnet: finds the object location with respect to anchor box if an object exists

4. Symbol detection: In this paper, 30 classes of different equipment's and valves have been explored to be thorough and exhaustive in effort to create an end to end algorithm. The challenge with these symbols was that they had very low interclass difference in their visual appearance, Thus standard off the shelf object detectors based on a traditional computer vision methods are not able to correctly distinguish between them as there is high intraclass variation, this prompted us to create a composite symbol detector based on Deep Learning methods.

Retina Net Architecture would be used for this task. Retina Net is a single, unified network made up of three individual networks, a single base network and two sub tasks specific the base network returns the feature mappings over the whole input, irrespective of the base network size and is a standard convolution network. The first subnet takes in the feature maps from the

first network and does the classification task which is probability of an object being present at a certain spatial location; the second subnet performs convolution bounding box regression using anchor boxes for each ground-truth object.

P&ID sheet symbols which are used as classes shown in Figure 4. Retina_Net Architexture is given in Figure 5.

Meta data and structuring: Now that all the P&ID components are extracted the final step of the algorithm is to create a mapping between each component and represent the P&ID digitally in a data format. This process is done as follows:

Pipeline Code to Pipeline Association: Here at this stage, pipeline and pipe-codes tags are associated. Each decision to associate any component with the other is done heuristically based on Euclidean Norm with respect to the distance between components, with a preference for components with the smallest distance.

Pipe code & tag association: Association of the inlet / outlet.

Pipe code & symbol association: Each component has its L2 Norm distance to all other components mapped and saved. The symbol and pipeline closest to each other based on L2 Norm are associated. The P&ID in its final form is represented as a random forest, each line tag is treated as a base node of the individual tree in the forest and the inlet tags are treated as leaves. Thus, each line would be intermediate nodes. At any given time, there can be trees with common nodes which show the line overlapping. However, each flow path is unique.

Table 1. An overview of experimental results

P&I Diagram Component	Detection Results	
	Detected vs Ground Truth	Accuracy
Pipeline-Code Detection	35 / 74	95.14%
Pipeline Detection	117 / 135	86.66%
Detected Outlets	23 / 24	95.88%
Detected Inlets	35 / 36	95.88%
Associating Pipeline Code	195 / 247	78.00 %
Associating Outlet	19 / 24	90.40%
Associating Inlet	33 / 36	94.11%

Experimental Results: These experiments were performed on a 16Gb Tesla V100 GPU & a private dataset for evaluating the performance of the proposed algorithm which was composed of a digital ledger which had about 1250 flow diagrams, was used. The experimental data was compiled in a tabular format. The experimental data was compiled in a tabular format and given in Table1.

In the above given table, the first entry shows the text detection accuracy given by the FOTS based proposed

OCR. 235 / 247 codes are positively found giving an accuracy of 95.14%. A sample visual output of the OCR Model is shown in fig 2. Next, the pipeline detection is handled, here lines are detected with an accuracy of 87%. The proposed heuristics for inlet and outlet-based detection works well giving 95% accuracy. Pipeline code association was at 78% as there is some noise in the data coupled with lack of detection of pipe codes or pipelines or sometimes both. Finally, for inlet and outlet association an average of 92% was found.

Table 2. Symbol detection confusion matrix for Retina Net

	VG	RC	VBF	VF	VC	RX	F	VG	VO	GI	VN
VG	79	5	0	0	0	0	0	4	0	0	0
RC	0	67	0	0	2	0	0	0	0	0	0
VBF	0	0	22	0	0	0	0	0	0	0	0
VF	0	0	0	394	0	0	0	0	0	0	0
VC	0	0	0	0	58	0	0	0	0	0	0
RX	0	0	0	0	0	24	0	0	0	2	0
F	0	0	0	0	0	6	16	0	0	9	0
VG	7	0	0	5	0	0	0	34	0	0	0
VO	0	0	0	0	0	0	0	0	261	0	0
GI	0	0	0	0	0	0	0	0	0	32	0
VN	0	0	8	0	0	0	0	0	3	0	129

Table 3. Retina Net performance metrics on different symbols

	Precision	Recall	F1-Score
VG	0.932	0.945	0.812
RC	0.956	0.941	0.955
VBF	0.99	0.883	0.944
VF	0.98	0.898	0.895
VC	0.97	0.812	0.897
RX	0.927	0.723	0.921
F	0.877	0.99	0.937
VG	0.999	0.922	0.836
VO	0.98	0.921	0.871
GI	0.99	0.789	0.893
VN	0.944	0.99	0.911

An error matrix is used to show the efficacy of symbol detection using Retina Net as seen above. The model is was trained for 8k epochs as pre-stoppage was employed to prevent overfitting coupled with use of simple overfitting prevention methods such as batch norm and data fuzzing. In order to cover maximum types of symbols there 10 unique classes of symbols to be detected, along with one extra class of spurious symbols was kept checking which classes needed extra data in order to get a boost in detection. This class of spurious symbols is used primarily to confuse the detector in order to get a better match in the long run. It has been seen experimentally that Retina Net model did not overfit due to these methods used. In some cases, as evidently seen that some classes have high intra class variation

and low interclass variation. Symbols such as ball valve, globe valve and globe valve (normally closed) as shown in fig are common examples of this phenomena. For example, 3 check valves are detected incorrectly, along with 8 flow arrows, and 4 globe valves. Symbols with high interclass variance such as reducer and flanges are detected without any false positives.

In Table 3, F1 score and the precision and recall values are calculated for symbol classes as seen in table 3. Retina

Net was found to have excellent F1 scores with symbols, even those with miniscule variations, for a score of 0.88 an average for each class. While in some cases symbols precision was recorded to be almost 1.0. Training loss along with overall loss, mean Average Precision coupled with regression loss and finally average class loss are given in Figure 6. Figure 7. shows a final color-coded digitized P&ID with connected component analysis.

Figure 6: Training loss along with overall loss, mean Average Precision coupled with regression loss and finally average class loss
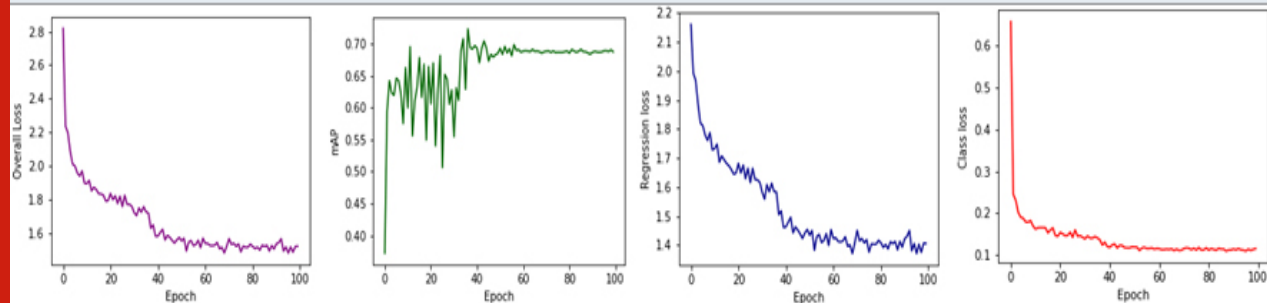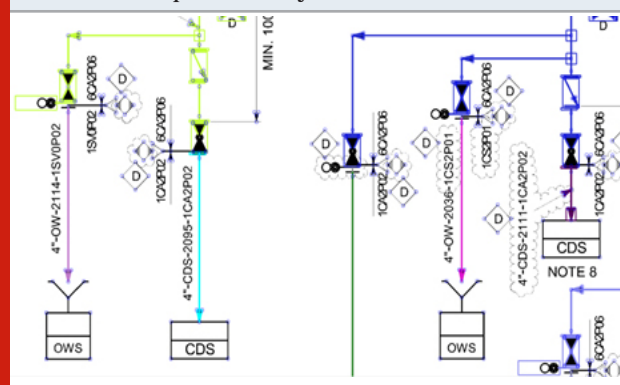

Figure 7: A final color-coded digitized P&ID with connected component analysis

CONCLUSION

In this paper, an algorithmic pipeline approach for P&ID data extraction and automatic digitizing, has been proposed. This proposed process combines current computer vision based and deep learning enabled models to recognize and classify symbols, semantically understand text and detect connections to link different equipment present on pipe runs. Additionally, the proposed symbol detection model is highly extensible and has a high level of precision and very low recall, therefore it can be easily trained on new classes. Geometric algorithms for instrument localization, traditional computer vision algorithms for detection of inlets, outlets and pipelines were used. Retina Net with a ResNet-101 backbone is used for pipeline code and symbol detection.

A random forest data structure is formulated for mapping each P&ID component and tracing the flow of fluid in

the appropriate pipeline. There are many directions which pursued here such as, using a 2 class CNN such as to classify whether a region of interest improves performance of symbol classifier in a binary manner as this reduces the sliding window computation. Secondly, one shot learning, leveraging a dataset made up of a handful of examples of each class could be done instead of assembling a dataset made up of hundreds of examples of a single class. The proposed heuristics for inlet and outlet-based detection work provides 95% accuracy. The symbol representations learned can be used in other places such as diagram search across projects and or automatically understanding the maintenance work needed across diagrams. New efficiencies can be found when unstructured data is converted into hierarchical structured data using the proposed pipeline, and benefit to industries is unlocked and the manual workload of documents digitization reduced.

REFERENCES

Adam, S., Ogier, J., Cariou, C., Mullot, R., Labiche, J., and Gardes, J. (2000). Symbol and character recognition: application to engineering drawings. International Journal on Document Analysis and Recognition.

Arroyo, E., Fay, A., Chioua, M., and Hoernicke, M. (2014). Integrating plant and process information as a basis for automated plant diagnosis tasks. In Proceedings of the 2014 IEEE Emerging Technology and Factory Automation (ETFA), pages 1–8.

Arroyo, E., Hoang, X. L., and Fay, A. (2015). Automatic detection and recognition of structural and connectivity objects in svg-coded engineering documents. In 2015 IEEE 20th Conference on Emerging Technologies Factory Automation (ETFA), pages 1–8.

Belongie, S., Malik, J., and Puzicha, J. (2002). Shape

matching and object recognition using shape contexts. IEEE Transactions on Pattern Analysis and Machine Intelligence, 24(4):509–522.

Verma, A., Sharma, M., Hebbalaguppe, R., Hassan, E., and Vig, L. (2016) Detecting text in natural image with connectionist text proposal network. CoRR, abs/1609.03605.

Dries Van Daele, Nicholas Decleyre, Herman Dubois, and Wannes Meert (2019). An automated engineering assistant: Learning parsers for technical drawings. ArXiv, abs/1909.08552, 2019.

Fei, L. and He, J. (2009). A three-dimensional douglaspeucker algorithm and its application to automated generalization of dems. International Journal of Geographical Information Science, 23(6):703–718.

Fu, L. and Kara, L. B. (2011). Neural network-based symbol recognition using a few labeled samples. Computers and Graphics, 35(5).

Gellaboina, M. K. and Venkoparao, V. G. (2009). Graphic symbol recognition using auto associative neural network model. In 2009 Seventh International Conference on Advances in Pattern Recognition, pages 297–301.

Goh, K. N., Mohd. Shukri, S. R., and Manao, R. B. H. (2013). Automatic assessment for engineering drawing. In Advances in Visual Informatics, pages 497–507, Cham. Springer International Publishing.

Gupta, G., Swati, Sharma, M., and Vig, L. (2017).

Information extraction from hand-marked industrial inspection sheets. In 2017 14th IAPR International Conference on Document Analysis and Recognition (ICDAR), volume 06, pages 33–38.

Kiryati, N., Eldar, Y., and Bruckstein, A. (1991). A probabilistic hough transform. Pattern Recognition, 24(4):303 – 316.

Koo, H. I. and Kim, D. H. (2013). Scene text detection via connected component clustering and nontext filtering. Trans. Img. Proc., 22(6):2296–2305.

Ouyang, T. Y. and Davis, R. (2009). A visual approach to sketched symbol recognition. In Proceedings of the 21st International Jont Conference on Artifical Intelligence, IJCAI'09, pages 1463–1468, San Francisco, CA, USA. Morgan Kaufmann Publishers Inc.

Ren, S., He, K., Girshick, R. B., and Sun, J. (2015). Faster R-CNN: towards real-time object detection with region proposal networks. CoRR, abs/1506.01497.

Saha, P. K., Borgefors, G., and di Baja, G. S. (2016). A survey on skeletonization algorithms and their applications. Pattern Recognition Letters, 76:3 – 12.

Sung-O Kang, Eul-Bum Lee, and Hum-Kyung Baek. (2019) A digitization and conversion tool for imaged drawings to intelligent piping and instrumentation diagrams (p&tid). Energies, 12(13):2593, Jul 2019.

Xuebo Liu, Ding Liang, Shi Yan, Dagui Chen, Yu Qiao, and Junjie Yan. (2018), FOTS: fast oriented text spotting with a unified network. CoRR, abs/1801.01671, 2018.