Universidade Federal do Paraná

Victor Duran Barroso

Deep Learning for Automatic Meter Reading

Curitiba PR
2018

Victor Duran Barroso

Deep Learning for Automatic Meter Reading

Thesis presented as a partial requirement to complete the bachelor degree in Computer Science.

Field: *Computer Science*.

Advisor: David Menotti.

Curitiba PR
2018

Universidade Federal do Paraná
Setor de Ciências Exatas
Curso de Ciência da Computação

Ata de Apresentação de Trabalho de Graduação II

Título do Trabalho: DEEP LEARNING FOR AUTOMATIC METER READING

Autor(es):

GRR__________ None: VICTOR DURAN BARROSO

GRR__________ Nome:

GRR__________ Nome:

Apresentação: Data: 04 / 07 / 2018 Hora: 9:45 Local: SALA DE REUNIÕES / DINF / UFPR

Orientador: DAVID MENOTTI GOMES

Membro 1: DANIEL WEINGAERTNER

Membro 2: EDUARDO TODT

(nome) (assinatura)

AVALIAÇÃO – Produto escrito		**ORIENTADOR**	**MEMBRO 1**	**MEMBRO 2**	**MÉDIA**
Conteúdo	(00-40)				40
Referência Bibliográfica	(00-10)				10
Formato	(00-05)				02
AVALIAÇÃO – Apresentação Oral					
Domínio do Assunto	(00-15)				15
Desenvolvimento do Assunto	(00-05)				03
Técnica de Apresentação	(00-03)				03
Uso do Tempo	(00-02)				02
AVALIAÇÃO – Desenvolvimento					
Nota do Orientador	(00-20)		***************	**************	20
NOTA FINAL		************	************	***********	95

Pesos indicados são sugestões.

Conforme decisão do colegiado do curso de Ciência da Computação, a entrega dos documentos comprobatório de trabalho de graduação 2 deve deve respeitar os seguintes procedimentos: Orientador deve abrir um processo no Sistema Eletrônico de Informações (SEI – UFPR); Selecionar o tipo: Graduação: Trabalho Conclusão de Curso; informar os interessados: nome do aluno e o nome do orientador; anexar esta ata escaneada e a versão final do pdf da monografia do aluno.; Tramita o processo para CCOMP (Coordenação Ciência da Computação).

I dedicate this for my parents, Helder and Marisol Barroso, which supported me through so many challenges, and for my advisor, David Menotti, for all his patience and diligence when guiding me.

Acknowledgements

I would like to give special thanks for my advisor, David Menotti, for all his efforts when guiding me on this work. Thanks to him, I could do work to be proud of and go through the personal challenges that came along.

I would like also to give special thanks Rayson Laroca for all his help in completing this work. This work is part of the results of a research paper written by the joint efforts of Rayson Laroca, Matheus A. Diniz, Gabriel R. Gonçalves and me, supervised by David Menotti and William Robson Schwartz. I'm grateful for having worked with them.

I thank my parents for their love, care and supporting me during all this time.

I'm also grateful for my university and all my professors for blessing me with their knowledge and experience so I could have the tools and confidence to reach my dreams. Thank you.

Abstract

Automatic reading of meters is a challenging task in the field of computer vision and scene text recognition that can benefit the very labor intensive process of consumption measurement of clients from electricity companies. In this paper, we propose an automatic meter recognition approach consisting of counter detection, digit segmentation and recognition. Also, we present the AMR dataset, containing 2000 images of different electricity meter models (10000 digits) taken from a local electricity company to be used as a standard benchmark. The method proposed was capable of 99.75% accuracy on counter detection while achieving 98.90% accuracy on individual digit recognition and 94.62% on counter recognition. An alternative approach for digit recognition was also tested and compared.

Keywords: deep learning, optical character recognition, electric meters.

Resumo

A leitura automática de medidores é uma tarefa desafiadora no campo da visão computacional e reconhecimento de texto de cena que pode beneficiar o processo muito trabalho intensivo processo de medição de consumo de clientes de empresas de eletricidade. Neste artigo, propomos uma abordagem de reconhecimento automático de medidores que consiste em detecção de contador, segmentação e reconhecimento de dígitos. Além disso, apresentamos o dataset AMR, contendo 2000 imagens de diferentes modelos de medidores de eletricidade (10000 dígitos) extraidas de uma companhia de eletricidade local para serem usadas como padrão de referência. O método proposto foi capaz de 99,75% de acurácia na detecção de contra-detecção, ao mesmo tempo em que atingiu 98,90% de acurácia no reconhecimento de dígitos individuais e 94,62% no reconhecimento de contadores. Um método alternativo para recohecimento de dígito também foi testado e comparado.

Palavras-chave: deep learning, reconhecimento óptico de caracteres, medidores elétricos.

Contents

List of Figures

List of Tables

List of Acronyms

UFPR	Universidade Federal do Paraná
UFMG	Universidade Federal de Minas Gerais
AMR	Automatic Meter Reading
ANN	Artificial Neural Network
CCA	Connected Components Analysis
CNN	Convolutional Neural Network
CRNN	Convolutional Recurrent Neural Network
FPS	Frames per Second
HOG	Histogram of Oriented Gradients
IoU	Intersection over Union
LSTM	Long Short-Term Memory
OCR	Optical Character Recognition
ROI	Region of Interest
LPR	License Plate Recognition
FCN	Fully Convolutional Network
MLP	Multilayer Perceptron
MSER	Maximally Stable Extremal Regions
STN	Spatial Transformer Network
SVM	Support Vector Machine
CTC	Connectionist Temporal Classification

Chapter 1

Introduction

The objective of this monograph is to propose a method using deep learning in both stages of counter detection and recognition to solve the task of Automatic Meter Reading (AMR). The topics that will be covered in this paper are:

- previous works in the field;
- present a new public dataset to be used as a benchmark for meter reading;
- propose a method for counter detection;
- propose and compare two methods for digit recognition;
- discuss the suitability of the methods to be used in mobile devices;

AMR refers to automatically record the consumption of electric energy, gas and water for both monitoring and billing. Despite the existence of smart readers [16], they are not widespread in many countries and the reading is still done manually on site by an operator, who takes a picture as reading proof [33, 10].

Since this operation is prone to errors, another operator needs to check the proof image to confirm the reading. This offline checking is expensive in terms of human effort and time, and has low efficiency [3]. Moreover, due to the large number of images to be evaluated, the audit is usually done by sampling [23] and errors might go unnoticed.

This audit being performed automatically would reduce mistakes introduced by human factor and save manpower. Furthermore, reading could also be done fully automatically through cameras installed in the meter box [30, 7]. Image-based AMR has advantages such as lower cost and quicker installation, since it does not require renewal or replacement of existent meters [35], and can ease the very labor intensive process of consumption measurement.

AMR based on images includes three phases, namely: (i) counter detection, (ii) digit segmentation and (iii) digit recognition. Counter detection is the fundamental stage, as its performance largely determines the overall accuracy and processing speed of the entire AMR system.

Deep Learning approaches are particularly dependent on the availability of large quantities of training data in order to generalize well and yield high classification accuracy on unseen data [27]. Some previous works on AMR (e.g., [10, 3]) employed huge datasets to train and evaluate their systems, however, these datasets were not made public. This is very common in AMR, since the images commonly belong to the [electricity, gas, water] company. In this sense, we introduce the AMR dataset, a new public dataset with 2000 images (i.e., 10000 digits), in order to assess the performance of the methods of each phase and provide means to compare with future works.

This work is part of a more comprehensive research paper written with the joint efforts of Rayson Laroca (UFPR), Victor D. Barroso (UFPR), Matheus A. Diniz (UFMG) and Gabriel R. Gonçalves (UFMG), supervised by David Menotti (UFPR) and William Robson Schwartz (UFMG).

This paper is organized as follows. We briefly review related work in Section chapter 2. The AMR dataset is introduced in chapter 3. Chapter 4 presents the proposed Convolutional Neural Network (CNN)-based approaches for AMR. We report and discuss the results in Section chapter 5. Conclusions and future work are given in chapter 6.

Chapter 2

Related Work

AMR intersects with other Optical Character Recognition (OCR) applications, such as robust reading and license plate recognition, as it must reliably extract text information from images taken under different conditions. Although AMR is not as widespread in the literature as these applications, a satisfactory number of works have been performed in recent years. Here a introduction about these works is presented. This section concludes with final remarks.

Many approaches exploited the vertical and horizontal pixel projections histograms for counter detection [37, 30, 7]. Projection-based methods can be easily affected by the rotation of the counter. Refs. [8, 23, 2, 10, 3] took advantage of prior knowledge such as counter's position and/or its colors (e.g., green background and red decimal digits). A major drawback of these techniques is that they might not work on all meter types and color information is not stable when the illumination changes. Other works include the use of template matching [23] and Adaboost [11]. In the latter, normalized gradient magnitude, Histogram of Oriented Gradients (HOG) and LUV color channels were adopted as features.

Projection and color-based approaches have also been widely employed for digit segmentation [8, 35, 19]. The use of morphological operations with Connected Components Analysis (CCA) was made in [3, 2], however, it depends largely on the result of binarization, as it cannot segment digits correctly if they are connected or broken. In [7], a binary digit/non-digit Support Vector Machine (SVM) was applied in a sliding window fashion, while Gallo et al. [10] exploited Maximally Stable Extremal Regions (MSER). In [10], the MSER algorithm failed to segment digits in images with problems such as blur and perspective distortion.

Nodari & Gallo [21] exploited an ensemble of Multilayer Perceptron (MLP) networks to perform the counter detection and digit segmentation without preprocessing and postprocessing stages. Since low F-measure rates were achieved, extra techniques were added in [33]. In summary, a watershed algorithm was applied to improve counter detection and Fourier analysis was employed to avoid false positives in digit segmentation. Although better results were attained, only 100 images were used to evaluate their system performance, which may not be representative enough.

Template matching [8, 19, 37] was widely used for digit recognition. Nevertheless, it is known that if a digit is different from the template due to any font change, rotation or noise, this approach produces incorrect recognition [6]. Therefore, many authors have employed an SVM classifier to this end. In [7, 33], simpler features such as pixel intensity were used in training, while HOG descriptors were adopted as features in [23, 10]. Lastly, the open-source Tesseract OCR Engine [31] was applied in [21, 3, 33], however, satisfactory results were not obtained in any of them.

AMR presents an unusual challenge in OCR: rotating digits. Typically, this is the major cause of errors, even when robust approaches are employed for digit recognition [36, 11]. In [19], this problem was addressed with an algorithm based on the Hausdorff distance, achieving excellent recognition results in real time. It should be noted, however, that all images were extracted from a single meter and a controlled environment was required since there were no preprocessing stage and no algorithm for angle correction.

Recently, deep learning approaches have won many pattern recognition competitions, even achieving superhuman visual results in some domains [28]. This motivated the use of deep learning for AMR, since only two works [3, 11] could be found employing CNNs in this context and both use private datasets and conventional image processing with handcrafted features in at least one stage. Moreover, (i) the images used in [11] are mostly sharp and very similar, which does not represent real-world conditions, and (ii) the poor digit segmentation accuracy obtained in [3], i.e. 81%, through a sequence of conventional image processing methods, prevents its use in real applications.

During the bibliographical research, only Nodari & Gallo [33] was found to have the datasets used in their experiments available. They proposed two datasets: one to evaluate the accuracy of their approach at each AMR stage and another to evaluate its overall accuracy. These datasets are composed of gas meter images with resolution around 640×480 pixels and the counter occupying a large portion in the image, which facilitates its location. Additionally, both datasets are very small (153 and 100 images) and the cameras used to capture them were not specified.

2.1 Final Remarks

The approaches developed for AMR are still limited. In addition to the aforementioned topics (i.e., private datasets and handcrafted features), many authors do not report the computational time of their approaches, making it impossible to examine its speed/accuracy trade-off, as well as its applicability. In this paper, for the first time, CNNs are used for both counter detection and recognition. In the proposed dataset, CNNs that achieved state-of-the-art results in other OCR applications were evaluated, reporting the computational time and the accuracy at digit/counter recognition level, in order to enable further comparisons.

Chapter 3

The AMR Dataset

The dataset contains 2,000 images taken from inside a warehouse of the Energy Company Of Paraná (Copel), which directly serves more than 4 million consuming units in the Brazilian state of Paraná [4]. Therefore, the dataset presents electric meters of different types and in different conditions. Fig. 3.2 shows the diversity of the dataset. One can see that the counter occupies a small portion in the image, which makes it difficult to locate it.

Meter images commonly have some artifacts (e.g., blur, reflections, low contrast, broken glass, dirt, among others) that may impair the reading of electric energy consumption. In addition, it is possible that the digits are rotating (e.g., between digits 4 and 5) in some kinds of counters. In such cases, the lowest digit was considered, since this is the protocol adopted at Copel. The exception is between digits 9 and 0, where it should be labeled as 9.

The images were acquired with three different cameras and are available[1] in the JPG format with resolution between 2,340 × 4,160 and 3,120 × 4,160 pixels. The cameras used were: *LG G3 D855*, *Samsung Galaxy J7 Prime* and *iPhone 6s*. As the cameras (cell phones) belong to different price ranges, the images presumably have different levels of quality. Additional information can be seen in Table 3.1.

Camera	**Images**
LG G3	948
J7 Prime	583
iPhone 6s	469
Total	2000

Info	**Counters**	**Digits**
Minimum Size	247 × 98	35 × 63
Maximum Size	1689 × 365	168 × 283
Average Size	682 × 180	76 × 134
Aspect Ratio	3.79	0.57

Table 3.1: Dataset information: (a) how many images were captured with each camera; (b) dimensions of meter counters and digits.

Note that, due to the nature of electric meters, the less significant digits (i.e., 0 and 1) have many more instances than the others (see Fig. 3.1). Nevertheless, digits 4-9 have a fairly similar number of examples.

The dataset is split into three sets: training (800 images), validation (400 images) and testing (800 images). This protocol was adopted (i.e., with a larger test set) to allow the reported results to be more statistically significant. This division was made randomly.

Every image has the following annotations available in a text file: the camera in which the image was taken, the counter's position and text, as well as the position of its digits. Each counter (regardless of meter type) has 5 digits, and thus 10,000 digits were manually annotated.

[1]The AMR dataset is publicly available to the research community at `https://web.inf.ufpr.br/vri/databases/aemr/`

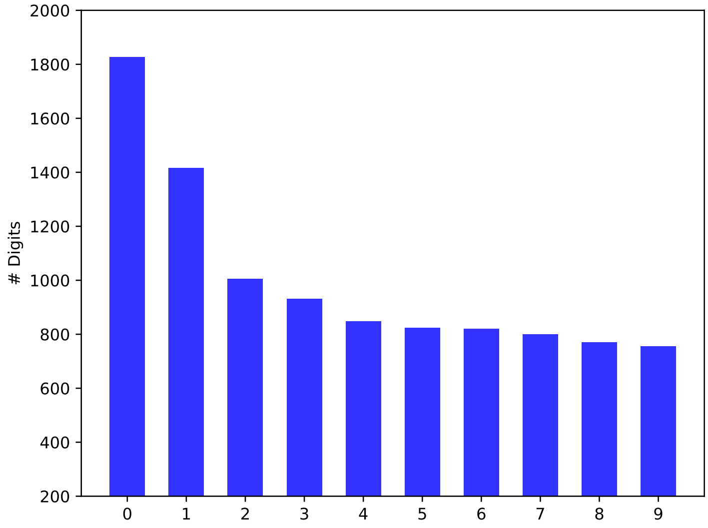

Figure 3.1: Frequency distribution of digits in the AMR dataset.

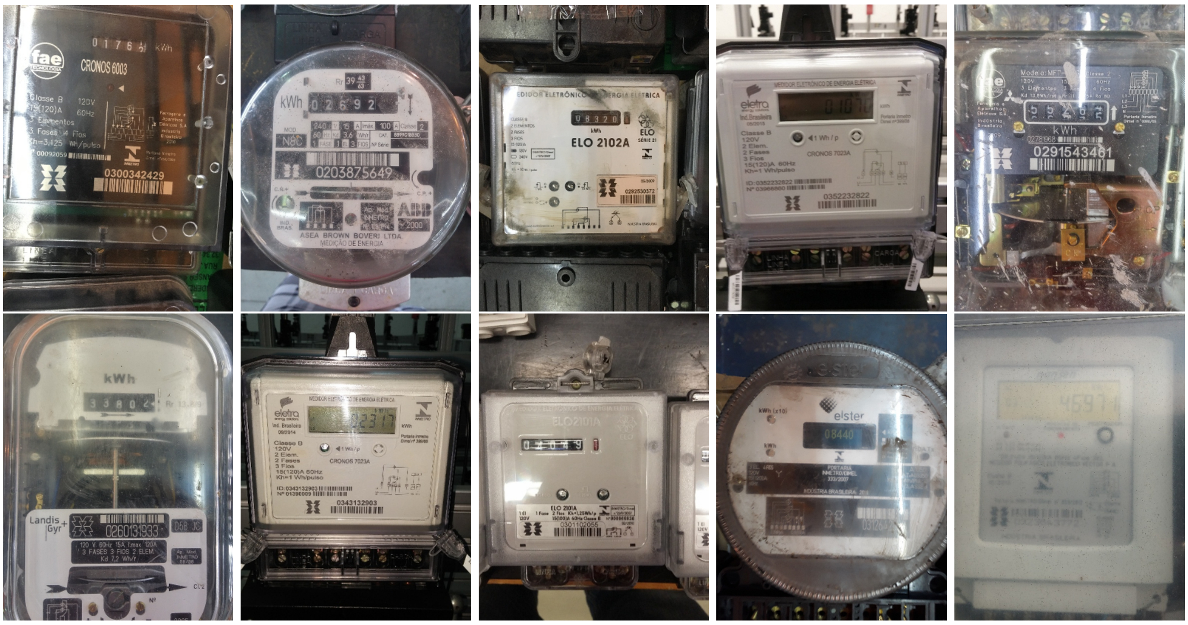

Figure 3.2: Sample images of the AMR dataset (some images were slightly resized for display purposes).

3.1 Final Remarks

The AMR dataset consists on 2000 electricity meter images (10000 digits) extracted using the default camera of 3 common types of smartphones at different resolutions. Distortion, digit rotation, partial occlusion of digits and small counter size are common in the dataset images, turning digit recognition into a very challenging task. All meters contain 5 digits, and, because the distribution frequency of digits is uneven, methods to solve the problem must account for this in order to avoid biased conclusions.

Chapter 4

Methodology

Electric meters have many textual information (e.g., meter specifications and serial number) that can be confused with the counter's text (i.e., the electric energy consumption). Moreover, the Region of Interest (ROI) usually occupies a small portion in the image. Therefore, first the counter region is located and then perform its recognition in the detected patch. both stages are tackled by leveraging the high capability of state-of-the-art CNNs.

This section describes the methodology and it is divided into two subsections: counter detection and counter recognition. It is worth noting that all parameters (e.g., CNNs input size, number of epochs, among others) are defined based on the validation set.

4.1 Counter Detection

Recently, great progress has been made in object detection through models inspired by YOLO [24, 34, 32], a CNN-based object detection system which regards detection as a regression problem. For that reason, it was decided to fine-tune it for counter detection. However, as only one class is required for detection and the computational cost is one of our main concerns, a smaller model was chosen, called Fast-YOLO [24], which uses fewer convolutional layers than YOLO and fewer filters in those layers. Despite being smaller, Fast-YOLO (architecture shown in Table 4.1) yielded great detection results in preliminary experiments.

Table 4.1: Fast-YOLO network used to detect the counter region.

	Layer	Filters	Size	Input	Output
0	conv	16	$3 \times 3/1$	$416 \times 416 \times 3$	$416 \times 416 \times 16$
1	max		$2 \times 2/2$	$416 \times 416 \times 16$	$208 \times 208 \times 16$
2	conv	32	$3 \times 3/1$	$208 \times 208 \times 16$	$208 \times 208 \times 32$
3	max		$2 \times 2/2$	$208 \times 208 \times 32$	$104 \times 104 \times 32$
4	conv	64	$3 \times 3/1$	$104 \times 104 \times 32$	$104 \times 104 \times 64$
5	max		$2 \times 2/2$	$104 \times 104 \times 64$	$52 \times 52 \times 64$
6	conv	128	$3 \times 3/1$	$52 \times 52 \times 64$	$52 \times 52 \times 128$
7	max		$2 \times 2/2$	$52 \times 52 \times 128$	$26 \times 26 \times 128$
8	conv	256	$3 \times 3/1$	$26 \times 26 \times 128$	$26 \times 26 \times 256$
9	max		$2 \times 2/2$	$26 \times 26 \times 256$	$13 \times 13 \times 256$
10	conv	512	$3 \times 3/1$	$13 \times 13 \times 256$	$13 \times 13 \times 512$
11	max		$2 \times 2/1$	$13 \times 13 \times 512$	$13 \times 13 \times 512$
12	conv	1024	$3 \times 3/1$	$13 \times 13 \times 512$	$13 \times 13 \times 1024$
13	conv	1024	$3 \times 3/1$	$13 \times 13 \times 1024$	$13 \times 13 \times 1024$
14	conv	30	$1 \times 1/1$	$13 \times 13 \times 1024$	$13 \times 13 \times 30$
15	detection				

For counter detection, we perform two minor changes in the Fast-YOLO[1] model. First, we recalculate the anchors for the AMR dataset using the algorithm available in [1]. Then, we reduce the number of filters in the last convolutional layer from 125 to 30 in order to output 1 class instead of 20. The number of filters in the last layer is given by

$$filters = (C + 5) \times A. \tag{4.1}$$

where A are the anchor boxes (we use $A = 5$) used to predict bounding boxes each with four coordinates (x, y, w, h) and confidence, and C are the class probabilities [25].

Fast-YOLO's multi-scale training was employed: every 10 batches the network randomly chooses a new image dimension size from 320×320 to 608×608 pixels. Then, we use 416×416 images as input since the best results (speed/accuracy trade-off in the validation set) were obtained with this dimension as input.

In cases where more than one counter is detected, only the detection with greater confidence was considered, since each image/meter has only one counter. Next, a margin (with size chosen based on the validation set) on the detected patch was added so that all digits are within the patch for the recognition stage. This is done to avoid losing digits in cases where the counter is not very well detected. A negative recognition result is given in cases where no counter is found.

4.2 Counter Recognition

Two CNN-based approaches were employed for counter recognition, since they presented good results in other OCR applications. These approaches are: CR-NET [20] and Convolutional Recurrent Neural Network (CRNN) [29]. The last doesn't require to go through the digit segmentation phase as it uses the whole counter image for digit recognition.

This chapter is divided into four parts, one to describe the data augmentation method, one for each CNN-based approach employed for counter recognition and one for final remarks regarding the methods used.

4.2.1 Data Augmentation

A straightforward algorithm was employed, proposed in [13], to generate new artificial images with counters that were not initially on the AMR dataset. The strategy consists on modifying the counter images by permuting the order of its digits. Thus, it is supposed that more examples that will help the CNNs to create an association of the digit position with the correspondent task.

Furthermore all images were converted to grayscale under the hypothesis that the noise generated by the image collage is reduced when compared to color images. The images generated are of lower resolution than the original, as this allows for smaller models with better run time while still achieving high recognition performance.

In order to account for the unbalance of digit classes for training, each counter position received the same number of different digits. Random variations of scale, rotation, translation and brightness were also performed in the augmented dataset. Some artificially generated images are shown in Fig. 4.1. As can be seen, the data augmentation approach works on different meter/counter kinds, creating new training examples for the CNNs.

[1]For training Fast-YOLO we used the weights pre-trained on ImageNet [5], available at `https://pjreddie.com/darknet/yolo/`

(a)

(b)

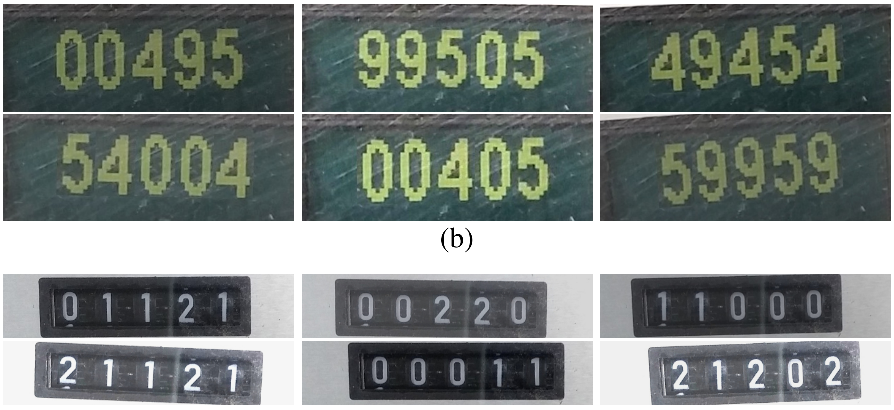

Figure 4.1: Some images created through data augmentation: the image in the upper left corner is the original one, while the others were generated automatically. (a) and (b) show that the same algorithm works for counters of different kinds and aspect ratios.

4.2.2 CR-NET

CR-NET is a YOLO-based model proposed for license plate character detection and recognition [20]. This model consists of the first eleven layers of YOLO and four other convolutional layers added to improve non-linearity. In [20], CR-NET (with input size of 240×80 pixels) was capable of detect and recognize license plate characters at 448 Frames per Second (FPS). Laroca et al. [18] also achieved great results applying CR-NET for this purpose.

The CR-NET architecture is shown in Table 4.2. As in the counter detection stage, the anchors for our data were recalculated and adjustments in the number of filters in the last layer were made. Furthermore, images with resolution of 400×106 pixels were chosen, since the results obtained when using other sizes (e.g., 360×95 and 440×116) were worse or similar, but with a higher computational cost. Note that the input image has the same aspect ratio of the counters (3.79, on average) in the AMR dataset.

Table 4.2: CR-NET with some changes for counter recognition: input size of 400×106 pixels and 75 filters in the last layer.

	Layer	Filters	Size	Input	Output
0	conv	32	$3 \times 3/1$	$400 \times 106 \times 3$	$400 \times 106 \times 32$
1	max		$2 \times 2/2$	$400 \times 106 \times 32$	$200 \times 53 \times 32$
2	conv	64	$3 \times 3/1$	$200 \times 53 \times 32$	$200 \times 53 \times 64$
3	max		$2 \times 2/2$	$200 \times 53 \times 64$	$100 \times 26 \times 64$
4	conv	128	$3 \times 3/1$	$100 \times 26 \times 64$	$100 \times 26 \times 128$
5	conv	64	$1 \times 1/1$	$100 \times 26 \times 128$	$100 \times 26 \times 64$
6	conv	128	$3 \times 3/1$	$100 \times 26 \times 64$	$100 \times 26 \times 128$
7	max		$2 \times 2/2$	$100 \times 26 \times 128$	$50 \times 13 \times 128$
8	conv	256	$3 \times 3/1$	$50 \times 13 \times 128$	$50 \times 13 \times 256$
9	conv	128	$1 \times 1/1$	$50 \times 13 \times 256$	$50 \times 13 \times 128$
10	conv	256	$3 \times 3/1$	$50 \times 13 \times 128$	$50 \times 13 \times 256$
11	conv	512	$3 \times 3/1$	$50 \times 13 \times 256$	$50 \times 13 \times 512$
12	conv	256	$1 \times 1/1$	$50 \times 13 \times 512$	$50 \times 13 \times 256$
13	conv	512	$3 \times 3/1$	$50 \times 13 \times 256$	$50 \times 13 \times 512$
14	conv	75	$1 \times 1/1$	$50 \times 13 \times 512$	$50 \times 13 \times 75$
15	detection				

Only the 5 digits detected/recognized with greater confidence were considered, since commonly more than 5 digits are predicted. However, it is worth noting that the same digit might be detected more than once by the network. Therefore, first, non-maximal suppression algorithm is applied to eliminate redundant detections. Although *highly unlikely*, it is also possible that less than 5 digits are detected by the CR-NET. In such cases, leading zeros were employed (e.g., 1234 → 01234).

4.2.3 Convolutional Recurrent Neural Network (CRNN)

The CRNN [29] model was designed for the scene text recognition task, which requires much more robustness than traditional OCR methods to handle a wide variety of fonts, backgrounds, lighting conditions, scales and angles.

This model consists of convolutional layers followed by recurrent layers, in addition to a custom transcription layer to convert the per-frame predictions into a label sequence. Given the cropped text (e.g., the counter), the convolutional layers act as a feature extractor, which is then transformed into a sequence of feature vectors and fed into a Long Short-Term Memory (LSTM) [12] recurrent layer. This layer handles the input as a sequence labeling problem, predicting a label distribution y_t for each frame $x = x_1, x_2, ..., x_t$. Since the LSTM is directional and the image-based sequences contain useful context information in both directions, two LSTMs are trained on the input sequence instead of one, the first on the input sequence as-is and the second on a reversed copy of the input sequence. Also, by stacking these bidirectional LSTMs, higher level of abstraction and significant better performance were attained in tasks such as speech recognition [14].

The Connectionist Temporal Classification (CTC) algorithm [15] is adopted for sequence decoding. Given the feature vectors $X = [x_1, x_2, ..., x_t]$ and the label sequence $Y = [y_1, y_2, ..., y_t]$, CTC first finds an accurate mapping from X to Y by calculating the conditional probability $P(Y|X)$ in the training phase. Then, CTC infers Y given an X by computing $Y* = argmax_Y P(Y|X)$. The CTC's objective function for a single (X, Y) is defined by

$$P(Y|X) = \sum_{A \epsilon A_{X,Y}} \prod_{t=1}^{T} P_t(a_t|X). \tag{4.2}$$

where $A_{X,Y}$ is the set of valid alignments. The advantage of CTC lies in the fact that it does not require an accurate alignment (correspondence of the elements) of X and Y.

Graves et al. [15] presented an efficient approximation for these conditional probabilities without the direct calculation of equation 4.2. The CRNN architecture is shown in Table 4.3.

The CRNN model was trained using PyTorch [22], with its default values for batch normalization (epsilon = 10^{-5}, momentum = 0.1) and adam optimization method [17] (lr = 10^{-3}, beta1 = 0.5, beta2 = 0.999, epsilon = 10^{-8}), mini-batch size of 128 images and 12 epochs. The input counter images were all resized to 40x160 pixels.

4.3 Final Remarks

The counters in the meter images are generally much smaller than the image itself, which requires a detection phase in the pipeline to extract the counter before the recognition phase, and the Fast-YOLO approach, commonly used for object recognition, was the method used to solve it. Two methods were tested for the counter recognition phase, named CR-NET and CRNN. The

Table 4.3: CRNN layers and hyperparameters.

Layer		Filters	Size	Input	Output
0	conv	64	$3 \times 3/1$	$40 \times 160 \times 1$	$40 \times 160 \times 64$
1	max		$2 \times 2/2$	$40 \times 160 \times 64$	$20 \times 80 \times 64$
2	conv	128	$3 \times 3/1$	$20 \times 80 \times 64$	$20 \times 80 \times 128$
3	max		$2 \times 2/2$	$20 \times 80 \times 128$	$10 \times 40 \times 128$
4	conv	256	$3 \times 3/1$	$10 \times 40 \times 128$	$10 \times 40 \times 256$
5	conv	256	$3 \times 3/1$	$10 \times 40 \times 256$	$10 \times 40 \times 256$
6	max		$2 \times 2/2 \times 1$	$10 \times 40 \times 256$	$5 \times 41 \times 256$
7	conv	512	$3 \times 3/1$	$5 \times 41 \times 256$	$5 \times 41 \times 512$
8	batch				
9	conv	512	$3 \times 3/1$	$5 \times 41 \times 512$	$5 \times 41 \times 512$
10	batch				
11	max		$2 \times 2/2 \times 1$	$5 \times 41 \times 512$	$2 \times 42 \times 512$
12	conv	512	$3 \times 3/1$	$1 \times 41 \times 512$	$1 \times 41 \times 512$
Layer		**Input Size**		**Hidden Layer Size**	**Output Size**
13	LSTM	512		256	256
14	LSTM	256		256	11
15	recognition				

CR-NET method requires a digit segmentation before digit recognition, while the CRNN uses the whole counter to perform the digit recognition.

To improve the performance in the digit recognition phase, data augmentation was performed in the cropped counters, generating a large number of images with permuted digits, greatly increasing the number of images to train the models.

Chapter 5

Experimental Results

In this chapter, experiments are performed to verify the effectiveness of the CNN-based methods in the AMR dataset. The experiments were implemented on a NVIDIA Titan XP GPU (3,840 CUDA cores and 12 GB of RAM). The YOLO-based models are trained using the Darknet framework [26].

First counter detection was evaluated, since the counters for recognition are from the detection results, rather than cropped directly from the ground truths. Thus, a well-performed counter detection is essential to achieve good recognition results.

5.1 Counter Detection

To evaluate counter detection, the bounding box evaluation defined in the PASCAL VOC Challenge [9] was employed, where the predicted bounding box is considered to be correct if its Intersection over Union (IoU) with the ground truth is greater than 50% (IoU ≥ 0.5). This metric was also used in previous works [21, 33], being interesting because penalizes both over- and under-estimates objects.

Some detection results achieved by the Fast-YOLO model are shown in Fig. 5.1. The network correctly detected 99.75% of the counters with a mean IoU of 0.83, failing to locate the counter in just two images (798/800). Actually, it is still possible to recognize the digits in these cases, since they were detected (with IoU ≤ 0.5) and all digits are within the ROI after adding a margin (as explained in Section 4.1). In the validation set, a margin of 20% (of the bounding box size) is required so that all digits are within the ROI. Thus, a 20% margin was applied in the test set as well. Fig. 5.2 shows both cases where the counters were detected with IoU ≤ 0.5 before and after adding this margin. Note that, in this way, the meter can be recognized even in poor counter detections.

In terms of computational speed, the Fast-YOLO model takes about 4.03 ms per image (248 FPS). The following parameters were used for training the network: $60k$ iterations (max batches) and learning rate = [1^{-3}, 1^{-4}, 1^{-5}] with steps at $25k$ and $35k$ iterations.

5.2 Counter Recognition

Here both counter and digit recognition accuracy are reported. The former is defined as the number of correctly recognized counters divided by the test set size, since each image has only one meter/counter. The latter is the number of correctly recognized digits divided by the number of digits in the test set. Additionally, all CNN models were trained with and without

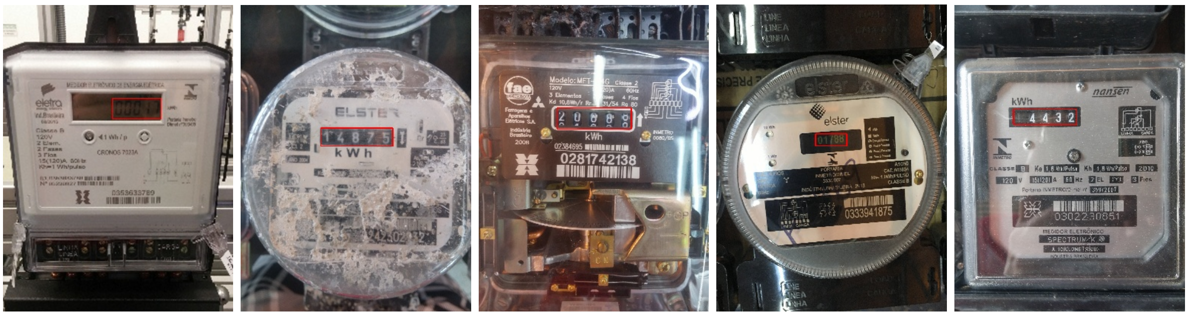

Figure 5.1: Samples of counter detection obtained with the Fast-YOLO model in the AMR dataset.

Figure 5.2: Bounding boxes predicted by the Fast-YOLO model before and after adding the margin (20% of the bounding box size).

data augmentation, and both results are reported. Thus, it is clear how data augmentation affects the performance of each model. Tables 5.1 and 5.2 summarize the accuracy and computational speed of the approaches.

Table 5.1: Accuracy CR-NET and CRNN with and without data agumentation

Approach	**Accuracy**	
	Digits	Counters
CR-NET (original training set)	98.26%	91.62%
CRNN (original training set)	96.82%	85.10%
CR-NET (with data augmentation)	98.66%	93.50%
CRNN (with data augmentation)	98.90%	94.62%

It is important to notice that a minor decrease in digit accuracy can greatly impact counter performance since its value is roughly translated to the power of 5 of the digit accuracy. Meter models with more digits can thus aggravate even more the performance of such systems.

Table 5.2: Time performance of CR-NET and CRNN on GPU

Approach	Time (ms)	FPS
CR-NET	4.03	248
CRNN	8.84	113

5.3 Final Remarks

As observed, the Fast-YOLO method has very high accuracy (99.75%) on dealing with the problem of counter detection on the AMR dataset, and its computational speed indicates being suitable to be implemented in mobile devices. For counter recognition, both CRNN and CR-NET possess very similar digit recognition accuracy (0.24% difference), but that translates to 1.12% difference in accuracy when dealing with counter recognition in favor to the CRNN method. That difference may be compensated by its computational speed being more than double that of the CRNN, making it better suited for mobile devices.

Chapter 6

Conclusion

In this work, the AMR dataset was described to serve as benchmark to new approaches to automatic meter reading. Also, a complete pipeline for the task of automatic meter reading was presented while analyzing two methods for the specific task of digit recognition. For the counter detection part, the Fast-YOLO reached nearly 100% accuracy (798 of 800 imagens) while managing to detect the ROI at 248FPS on GPU, being a model suitable for handheld devices. For digit recognition, the CRNN performed a marginaly better than the CR-NET (94.62% as compared to $93,5\%$) while being twice as computationally intensive.

The system was designed mainly to deal with the electricity meter models present in the AMR dataset, but it can be easily modified to accommodate new models (including water and gas meters), being the number of examples the main constraint. Since the AMR dataset contains only meters with 5 digits, small changes in the architecture may be required to be used along with the CR-NET method. The CRNN, being a sequence-based method, is unconstrained by the number of outputs, so it can deal with meter models with different number of digits without changes in the architecture.

The automatic meter reading task is still an open problem in the field of computer vision and scene text recognition. A variety of meters, including many old models, are in use around the globe, an the system must account for these variations in order to be effectively effectively deployed. Pointer meter readers are also commonly used, and specific computer vision methods to extract the meter information must be developed to be used along with the methods described in this paper. Further works also include evaluating performance in handheld devices with different hardware and configurations, refine the methods for digit recognition and compare such methods with human performance.

Bibliography

[1] AlexeyAB. YOLOv2 and YOLOv3: how to improve object detection. https://github.com/AlexeyAB/darknet#how-to-train-to-detect-your-custom-objects, 2018. Accessed: 2018-06-14.

[2] A. Anis, M. Khaliluzzaman, M. Yakub, N. Chakraborty, and K. Deb. Digital electric meter reading recognition based on horizontal and vertical binary pattern. In *Int. Conference on Electrical Information and Communication Technology*, pages 1–6, Dec 2017.

[3] Martin Cerman, Gayane Shalunts, and Daniel Albertini. A mobile recognition system for analog energy meter scanning. In *Advances in Visual Computing*, pages 247–256. Springer International Publisher, 2016.

[4] Copel. Energy Company Of Paraná. `http://www.copel.com/hpcopel/english/`, 2018. Accessed: 2018-04-24.

[5] J. Deng, W. Dong, R. Socher, L. J. Li, Kai Li, and Li Fei-Fei. ImageNet: A large-scale hierarchical image database. In *Conference on Computer Vision and Pattern Recognition*, pages 248–255, June 2009.

[6] S. Du, M. Ibrahim, M. Shehata, and W. Badawy. Automatic license plate recognition (ALPR): A state-of-the-art review. *Trans. on Circuits and Systems for Video Technology*, 23(2):311–325, Feb 2013.

[7] V. C. P. Edward. Support vector machine based automatic electric meter reading system. In *International Conference on Computational Intelligence and Computing Research*, pages 1–5, Dec 2013.

[8] L. A. Elrefaei, A. Bajaber, S. Natheir, N. AbuSanab, and M. Bazi. Automatic electricity meter reading based on image processing. In *IEEE Jordan Conference on Applied Electrical Engineering and Computing Technologies (AEECT)*, pages 1–5, Nov 2015.

[9] Mark Everingham, Luc Van Gool, Christopher K. I. Williams, John Winn, and Andrew Zisserman. The pascal visual object classes (VOC) challenge. *International Journal of Computer Vision*, 88(2):303–338, Jun 2010.

[10] I. Gallo, A. Zamberletti, and L. Noce. Robust angle invariant GAS meter reading. In *International Conference on Digital Image Computing: Techniques and Applications*, pages 1–7, Nov 2015.

[11] Yunze Gao, Chaoyang Zhao, Jinqiao Wang, and Hanqing Lu. Automatic watermeter digit recognition on mobile devices. In *Internet Multimedia Computing and Service*, pages 87–95. Springer Singapore, 2018.

[12] F. A. Gers, J. Schmidhuber, and F. Cummins. Learning to forget: continual prediction with LSTM. In *International Conference on Artificial Neural Networks*, volume 2, pages 850–855 vol.2, 1999.

[13] Gabriel Resende Gonçalves, Matheus A. Diniz, Rayson Laroca, David Menotti, and William Robson Schwartz. Low-resolution license plate recognition by multi-task deep learning. *CoRR*, abs/xxxx.xxxxx, 2018.

[14] A. Graves, A. r. Mohamed, and G. Hinton. Speech recognition with deep recurrent neural networks. In *International Conference on Acoustics, Speech and Signal Processing*, pages 6645–6649, May 2013.

[15] Alex Graves, Santiago Fernández, Faustino Gomez, and Jürgen Schmidhuber. Connectionist temporal classification: Labelling unsegmented sequence data with recurrent neural networks. In *International Conference on Machine Learning (ICML)*, pages 369–376, 2006.

[16] Yasin Kabalci. A survey on smart metering and smart grid communication. *Renewable and Sustainable Energy Reviews*, 57:302 – 318, 2016.

[17] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *CoRR*, abs/1412.6980, 2014.

[18] Rayson Laroca, Evair Severo, Luiz A. Zanlorensi, Luiz S. Oliveira, Gabriel Resende Gonçalves, William Robson Schwartz, and David Menotti. A robust real-time automatic license plate recognition based on the YOLO detector. *CoRR*, 2018.

[19] D. Jabba M. Calle M. Rodriguez, G. Berdugo and M. Jimeno. HD MR: a new algorithm for number recognition in electrical meters. *Turkish Journal of Elec. Engineering & Comp. Sciences*, 22:87–96, Jan 2014.

[20] S. Montazzolli and C. R. Jung. Real-time brazilian license plate detection and recognition using deep convolutional neural networks. In *30th Conference on Graphics, Patterns and Images (SIBGRAPI)*, pages 55–62, Oct 2017.

[21] Angelo Nodari and Ignazio Gallo. A multi-neural network approach to image detection and segmentation of gas meter counter. In *IAPR Conference on Machine Vision Applications*, pages 239–242, 2011.

[22] Adam Paszke et al. Automatic differentiation in PyTorch. 2017.

[23] D.B.P. Quintanilha et al. Automatic consumption reading on electromechanical meters using HoG and SVM. In *Latin American Conf. on Networked and Electronic Media*, pages 11–15, Nov 2017.

[24] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi. You only look once: Unified, real-time object detection. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 779–788, June 2016.

[25] J. Redmon and A. Farhadi. YOLO9000: Better, faster, stronger. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 6517–6525, July 2017.

[26] Joseph Redmon. Darknet: Open source neural networks in C. `http://pjreddie.com/darknet/`, 2013–2018.

[27] J. Salamon and J. P. Bello. Deep convolutional neural networks and data augmentation for environmental sound classification. *IEEE Signal Processing Letters*, 24(3):279–283, March 2017.

[28] Jürgen Schmidhuber. Deep learning in neural networks: An overview. *Neural Networks*, 61:85 – 117, 2015.

[29] B. Shi, X. Bai, and C. Yao. An end-to-end trainable neural network for image-based sequence recognition and its application to scene text recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 39(11):2298–2304, Nov 2017.

[30] D. Shu, S. Ma, and C. Jing. Study of the automatic reading of watt meter based on image processing technology. In *IEEE Conf. on Industrial Electronics and Applications*, pages 2214–2217, May 2007.

[31] R. Smith. An overview of the Tesseract OCR Engine. In *International Conference on Document Analysis and Recognition*, volume 2, pages 629–633, Sept 2007.

[32] S. Tripathi, G. Dane, B. Kang, V. Bhaskaran, and T. Nguyen. LCDet: Low-complexity fully-convolutional neural networks for object detection in embedded systems. In *IEEE Conf. on Comp. Vision and Pattern Recognition Workshops*, pages 411–420, July 2017.

[33] Marco Vanetti, Ignazio Gallo, and Angelo Nodari. Gas meter reading from real world images using a multi-net system. *Pattern Recognition Letters*, 34(5):519–526, 2013.

[34] Bichen Wu, Alvin Wan, Forrest Iandola, Peter H. Jin, and Kurt Keutzer. SqueezeDet: Unified, small, low power fully convolutional neural networks for real-time object detection for autonomous driving. *CoRR*, 2016.

[35] Yunzhou Zhang, Shanbao Yang, Xiaolin Su, Enyi Shi, and Handuo Zhang. Automatic reading of domestic electric meter: an intelligent device based on image processing and ZigBee/Ethernet communication. *Journal of Real-Time Image Processing*, 12(1):133–143, Jun 2016.

[36] L. Zhao, Y. Zhang, Q. Bai, Z. Qi, and X. Zhang. Design and research of digital meter identifier based on image and wireless communication. In *International Conference on Industrial Mechatronics and Automation*, pages 101–104, May 2009.

[37] Shutao Zhao, Baoshu Li, Jinsha Yuan, and Guiyan Cui. Research on remote meter automatic reading based on computer vision. In *IEEE/PES Transmission Distribution Conf. Expo.: Asia and Pacific*, pages 1–4, 2005.