Leakage Detection in District Heating Systems Using UAV IR Images: Comparing Convolutional Neural Network and ML Classifiers

Kabir Hossain
Department of Photonics Engineering
Technical University of Denmark
Kongens Lyngby, Denmark
kabha@fotonik.dtu.dk

Frederik Villebro
Drone Systems
Århus, Denmark
frederik@dronesystems.dk

Søren Forchhammer
Department of Photonics Engineering
Technical University of Denmark
Kongens Lyngby, Denmark
sofo@fotonik.dtu.dk

***Abstract*—In this paper, we proposed a method to detect leakages automatically in underground pipes of district heating networks based on images, which are captured by an Unmanned Aerial Vehicle (UAV). The original datasets are captured in a 16 bits format and later converted into an 8 bit format using Dynamic Range Reduction (DRR). Leakages in district heating networks can occur due to unprofessional installation, lack of maintenance or end of service life, etc. We have addressed issues of leakage detection using a deep learning based approach, Convolutional Neural Network (CNN), and 8 machine learning classifiers.**

The experiments are carried out on seven different datasets, which are acquired at seven different cities in Denmark. We performed our experiments on both 16 bits and 8 bits data. For performance analysis, 6 datasets are used for training and the remaining dataset for testing. Our proposed deep learning CNN achieves an average accuracy of 0.886 and 0.884 for 16 bits and 8 bits, respectively. Machine learning classifiers such as Adaboost (AB), Random Forest (RF) etc provide relatively lower average accuracy. Adaboost required less computational resources, achieves average accuracies of 0.800 and 0.793 for 16 bits and 8 bits, respectively.

***Index Terms*—Convolution Neural Networks, SVM, RF, Adaboost, Leakage detection, district heating system**

I. Introduction

District heating systems supply heat through underground pipes, which carry hot water from a central power plant. This is the most common and environmental friendly [1] way to provide heat in northern countries. Heat leakages are a common problem, because pipes degenerate with time [2] for many reasons, e.g. due to unprofessional installation, lack of maintenance or end of service life. The performance decreases due to the leakages and this creates a negative impact on the environment [3]. Therefore, it is required to monitor the networks regularly to detect energy leakages.

Currently all infrared images are reviewed and leakages localized manually. This is an extremely time-consuming task and in most cases the bottle neck in the surveying process. Full automation of leakage detection or at least a reduction in images to review would dramatically decrease the time required increasing productivity and scalability. Furthermore, it is possible that Machine Learning (ML) models are able to spot earlier signs of leakages compared to a human operator resulting in prevention rather than intervention. Therefore, in this paper, we focus on leakage detection of district heating system using ML.

Over the years, researchers have worked on monitoring the energy leakages using various methods. The manual way is to measure the flow of water at the inlet and outlet. A leakage is considered based on differences between inlet and outlet flow. However the inlet-outlet measurement is less precise in the location of the leakage. There is another method in which fiber sensors are installed within the pipes to detect leakages automatically. However, the fiber sensors require further installation and is rather intrusive to the system. As an advanced technique, thermal imaging has been introduced in the last few decades to detect the leakage efficiently [4]–[7]. B. Bøhm et al. [4], present a method for energy leakage detection using hand-held cameras. This technique has many drawbacks since it has less scalability and it is difficult to monitor many areas compared with aerial thermography.

Aerial thermography is applied for energy leakage detection. In [5]–[7], regions of interest were extracted by orthorectification and using information about the pipe location. Furthermore, a list of features such as mean, standard deviation, circularity etc. were defined manually. Thereafter, five conventional ML classifiers were used to evaluate the performance. Moreover, their hardware setup is more costly, since they utilized a plane and a cooled thermal camera compared to a UAV with an uncooled thermal camera as in our case. For our target application, we provide a relatively cheap and simple UAV based solution to detect leakages in pipes of district heating networks. Since we are using UAV, this is non-intrusive and tells the precise location of the leakages. We have applied CNN for leakage detection and the motivation behind the CNN model is that it can take two dimensional data directly as input without requiring prior feature engineering. However, the remaining conventional ML classifiers utilize the dense SIFT and SURF features [8] for training the model.

In our setup, the IR sensor is embedded in the UAV and the

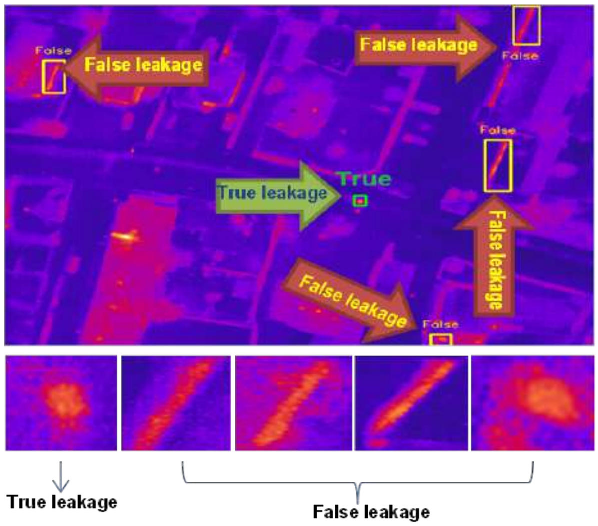


Fig. 1: IR image of district heating energy inspection captured by UAV

IR images are captured on-board. The images are captured in 16 bit format, then converted into 8 bit format using a histogram based DRR operator. Thereafter, potential leakages are extracted from IR images using a region extraction algorithm. The ground truth values are identified by a human observer. These potential leakages and their corresponding labels are used for training CNN and conventional ML classifiers to classify into true and false leakages. Examples of the true and false leakages are shown in Figure 1.

The key contribution of this work is that we provide a framework to detect the leakages of district heating networks using conventional ML classifiers and a CNN model based on UAV images. The challenge is that there are many more signatures than actual leakages. In our work, the ML classifiers and CNN are applied on raw 16 bit images directly to detect leakages, optionally ML is also applied on 8 bit images. It should be noted that 16 bit images contain 65536 gray levels, which give detailed information. Secondly, our CNN model provides higher accuracy with lower false positive rate. In contrast, conventional ML classifiers gave reasonable results but required less computational resources. Finally, we provide a relatively simple and cheap UAV based solution for monitoring district heating networks.

The remainder of this paper is organized as follows: The proposed method is detailed in Section II. In Section III, experimental results are presented: description of district heating data sets, and performance of the proposed method along with the discussion.

II. Methodology

We consider detecting the leakages of district heating networks using conventional Machine Learning (ML) and Convolutional Neural Network (CNN) classifiers. In this section the image pre-processing is briefly described. For conventional ML classifiers, the features are extracted from input images and then trained on. On the other hand, CNN is directly fed with the input images and corresponding labels for training. In this Section, we describe the conventional ML and CNN classifiers.

TABLE I: Parameters of Conventional ML Classifiers

Methods	Parameters
RF	Number of estimators = 200
Adaboost	Number of estimators = 200
	Weak classifier: Decision stump
	Learning rate = 1
SVM	Penalty parameter C = 1
	Kernel = 'Linear'

A. *Image Pre-processing*

We performed pre-processing before feeding the data into the machine learning algorithms. It should be noted here that leakages appear as hot in IR images, but not all hot / warmer regions are real leakages. In our scenario, an UAV captures IR images of the area with pipelines being surveyed. After capture an expert reviews the images for potential leakages and classifies according to severity. Severity is estimated by looking at the temperature differences as well as the surroundings and requires some training as the minor leakages can be very subtle. The potential leakages are then located and rectified if necessary. In addition, we developed a region extraction algorithm to extract the image patches (both in 8 and 16 bits), containing both the true and false leakages. The patches not classified as real leakages by the human observer are labelled as false leakages for training purpose.

The input image patches are then rescaled into 70 x 70 for working with ML classifiers and CNN. It should be noted here, original data should not be directly input to the conventional ML classifiers and CNN. It is recommended to do normalization, because it makes samples less sensitive to small intensity changes and ensure faster convergence of the CNN. The simple normalization is:

$$y = (x - mean)/stddev, \tag{1}$$

where x and y are the pixel values, $stddev$ is the standard deviation and $mean$ is the mean value of image pixels.

For CNN, the storage structure is established based on the format of the MNIST database of handwritten digits [9], which includes headers, labels and pictures.

B. *Conventional ML Algorithms*

For evaluation, we investigated eight ML classifiers. In order to train with the ML classifiers, feature descriptors are computed.

1) Features: Features are computed based on dense sampling, where the image is subdivided into n x n grid points. The grid size used in our implementation is 20 x 20. Then we extract SIFT and SURF descriptors from each grid point as in [8]. In contrast to the conventional SIFT and SURF descriptors, the number of features does not vary with image content. We have adapted the implementation of dense sampling as described in [8]. A total of four key points per

TABLE II: 7 district heating image datasets

Dataset	S1	S2	S3	S4	S5	S6	S7	Total
Number of True	557	1182	2346	5392	957	1327	1665	13426
Number of False	540	1200	2400	5420	992	1372	1700	13624
Total # of sample	1097	2382	4746	10812	1949	2699	3365	27050

TABLE III: Accuracy of 9 machine learning classifier algorithms for performance analysis

Datasets	Bits	KNN	DT	AB	RF	LR	L-SVM	LDA	GNB	CNN
S1	16	0.74	0.73	0.83	0.82	0.80	0.80	0.78	0.69	**0.94**
	8	0.75	0.75	0.85	0.83	0.82	0.82	0.81	0.72	**0.93**
S2	16	0.74	0.73	0.80	0.79	0.79	0.78	0.79	0.74	**0.86**
	8	0.76	0.70	0.78	0.79	0.77	0.77	0.77	0.70	**0.86**
S3	16	0.76	0.73	0.81	0.81	0.78	0.78	0.77	0.71	**0.90**
	8	0.75	0.71	0.80	0.79	0.78	0.78	0.77	0.72	**0.90**
S4	16	0.76	0.74	0.83	0.85	0.81	0.80	0.79	0.76	**0.90**
	8	0.75	0.72	0.83	0.83	0.81	0.80	0.79	0.75	**0.90**
S5	16	0.73	0.70	0.80	0.79	0.76	0.77	0.76	0.71	**0.89**
	8	0.72	0.69	0.77	0.78	0.77	0.77	0.75	0.69	**0.85**
S6	16	0.69	0.64	0.68	0.68	0.69	0.69	0.70	0.63	**0.76**
	8	0.69	0.63	0.70	0.68	0.69	0.69	0.70	0.62	**0.84**
S7	16	0.76	0.75	0.85	0.84	0.82	0.81	0.81	0.72	**0.93**
	8	0.74	0.75	0.82	0.83	0.81	0.82	0.80	0.76	**0.91**

patch are found using SIFT and SURF. From these four key points, we extract a total of 512 and 256 features for SIFT and SURF, respectively.

After feature extraction from each grid point using SIFT and SURF, we have combined these two descriptors indexed into a HDF5 [10] file format.

2) ML Classifiers: In our work a set of eight ML classifiers are tested for evaluation. Four of them are linear and the other four are non linear classifiers. The linear models used in this work are Logistic Regression (LR) [11], Linear Discriminant Analysis (LDA) [12], Linear Support Vector Machine (L-SVM) [13] and Gaussian Naive Bayes (GNB) [14].

The four non-linear classifiers are k-nearest neighbors (KNN) [15], Decision Tree(DT) [16], Random Forest (RF) [17] and Adaboost(AB) [18]. The parameters we used for RF, Adaboost and SVM can be seen in Table I. It should be noted that our AB classifier uses decision stumps [19] as it is a weak learner equivalent to a decision tree with a maximum depth of 1.

C. Architecture of Convolutional Neural Network (CNN)

The CNN takes the 70x70 input and feeds it through 3 convolution layers followed by 2 fully connected layers. The first 2 convolutional layers have a kernel of 3x3 with 64 filters each. The third convolutional layer has the same kernel size but 128 filters. It should be noted that each convolutional layer is utilizing the leaky ReLu activation function with a rate of 0.1. Each convolution layer is also followed by a maxpooling layer with a kernel of 2x2 and a dropout layer with a rate of 0.25 and 0.5 for the first 2 layers and the last 2 layers, respectively. The first fully connected layer has a total of 128 neurons while the second has 2 responsible for classification.

The optimizer used in this structure is Adam [20] and the loss function is cross entropy. The epoch is defined as 20 with batch sizes of 64. The CNN is implemented in Python using Keras [21] library. The reason we have chosen this CNN architecture is that this is for proof of concept and this structure gave reasonable results. In the future, we can consider more advanced deep learning architecture such as ResNet.

III. EVALUATION

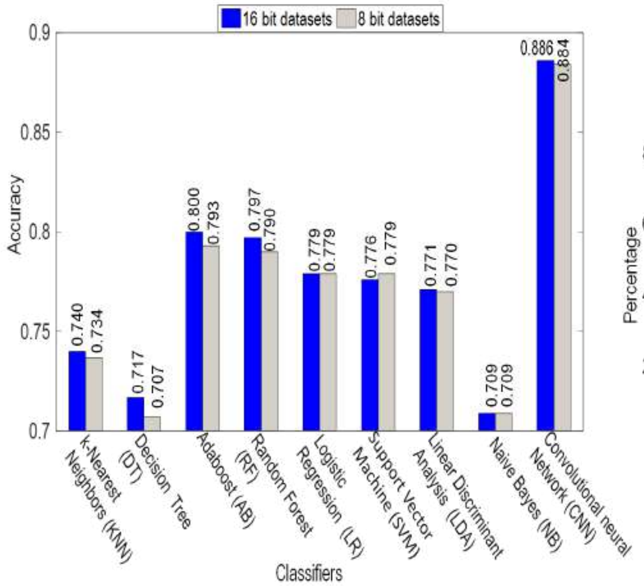
For performance analysis, the Deep learning approach using CNN was used to detect leakages of district heating system images with high accuracy. A set of features are also extracted using SIFT and SURF descriptors. Thereafter, the machine learning classifiers LR [11], LDA [12], KNN [15], DT [16], RF [17], GNB [14], SVM [13] and AB [18] were applied. Performances was evaluated by various performance parameters, i.e. True Positive Rate (TPR), False Positive Rate (FPR) and accuracy, which is calculated as the number of correct matches between ground truth and predictions, divided by the number of samples, etc.

In this Section, we assess the performance of our proposed CNN model and the conventional ML classifiers applied to leakage detection.

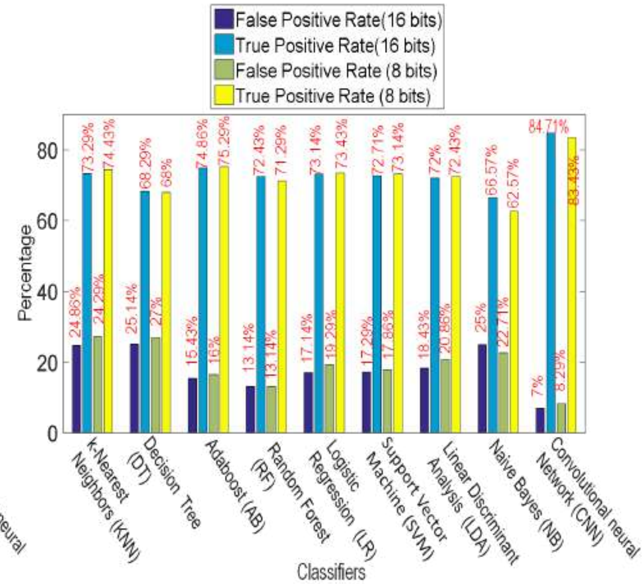
A. Dataset

We have used district heating system data, which is provided by Drone Systems (Denmark). The dataset contains a total of 7 IR sequences acquired by a UAV and each of the sequences is captured at one of seven different cities in Denmark. The UAV sequences (S1-S7), originally available in 16 bits, are also converted into 8 bit format by histogram based Dynamic Range Reduction (DRR). The experiments are carried out on both the 16 and 8 bit sequences.

Later, we have extracted image patches at a resolution of 70 x 70 pixels from the 7 different UAV sequences. In total, we extracted 27050 image patches, where 13426 contain true leakages and the other 13624 image patches are false leakages. The detailed number of image patches extracted from each dataset can be found in Table II.



(a) Average accuracy



(b) True positve vs false positive

Fig. 2: Average accuracy and corresponding true and false positive rate of different ML classifiers

B. Performance Analysis

For experiments, we have used 6 datasets (out of 7) for training, and the remaining dataset for testing performed on all splits of datasets for performance analysis. Leave-one-out was applied for both CNN and the ML classifiers. In Table III, the experimental results are shown for each of the datasets. We calculate the average value over all sequences as shown in Fig. 2(a).

As shown in Figure 2 (a), the best results are obtained by Deep learning based CNN, achieving an accuracy of 0.886 and 0.884 for 16 bits and 8 bits, respectively. Even the corresponding true and false positive rate is also far better than the conventional ML classifiers as shown in Figure 2(b). For CNN, the true reported positive rates are 84.71% and 83.43%, and the false positive rates are 7.00% and 8.29% for 16 bits and 8 bits, respectively.

Among the ML classifiers, the non-linear AB and RF provide the best results. The AB average accuracy values obtained are 0.800 and 0.793 for 16 bits and 8 bits, respectively. The corresponding true positive rates are 74.86% and 57.00%, false positive rates are 15.43% and 16.00% for 16 bits and 8 bits, respectively. The average accuracy obtained for RF are 0.797 and 0.790 for 16 bits and 8 bits, respectively. The corresponding true positive rates are 72.43% and 71.29%, false positive rates are 13.14% and 13.14% for 16 bits and 8 bits, respectively.

Overall the 16 bits achieved slight better results compared to 8 bits images in terms of accuracy and true positive rate.

C. Discussion

Figure 2 shows that the Convolutional Neural Network (CNN) gave the best results and second best results are obtained by Adaboost (AB) and Random Forest (RF) in terms of accuracy, thereafter comes Linear Regression (LR), Support Vector Machine (SVM), and Linear Discriminant Analysis (LDA) all being very similar in performance. It could be noted that the Decision Tree (DT) and Naive Bayes (NB) gave the relatively poorest results among all ML algorithms.

Our CNN model with the second best classifiers, i.e. AB, accuracy increases by 10.75% and 11.48% for 16 and 8 bits, respectively. The performance difference between CNN and conventional methods might be explained by the fact that the former utilizes the raw input, while the latter relies on features extracted from a 20x20 grid.

It should be noted, higher true positive rate means lower false negative rate, and vice versa. Higher false negative rate, i.e. overlook a leakage is more expensive than higher false positive rate in our target application. Considering this fact, CNN satisfy the above criteria. CNN gave higher detection accuracy, but it requires more computational resources. In contrast, conventional ML classifiers, which are relatively simple and fast, require less computational power. Thus from a real time perspective, conventional ML classifiers could be more effective.

We have compared our results with the results in [6]. As described above (See Section I) Berg et al. [6], present the result for energy leakage detection using a plane. In their work, they have used up to 50,000 thermal images and stitched them for energy leakage detection. Later they have extracted the patches and applied list of machine learning classifiers (LDA, L-SVM, RBF-SVM, RF, and AB) on it, where the best result is found using RF. They achieved a true positive rate of 99%, while the false positive rate is around 42%. For real application on the drone data, this high FPR will vastly dominate compared with the true leakages. In contrast, we achieved a much lower false positive rate of only 7%, which is much lower than them, while our true positive rate is around 84.71%. It should be noted, they have one chance to detect leakage since there are using stitching image, but in our case, we have analyzed each of the images individually. Thus a leakage is not required to be detected on all the individual images. Processing across multiple images can increase the success rate of capturing a given leakage present in several images. However, this is left for further study.

IV. CONCLUSIONS

In this paper, we have presented a leakage detection method for district heating systems using CNN and machine learning classifiers. The original UAV image sequences are 16 bits, then converted in 8 bit format using a DRR operator. Afterwards, the image patches are extracted by using a region extraction algorithm. For ML classifiers, input images are divided in fixed grid points and then a set of SIFT and SURF descriptors are extracted for training and testing with machine learning techniques. For Deep learning CNN, input samples are directly fed into the network for leakage detection.

Using a leave-one-out approach, 6 datasets are used for training and the remaining dataset for testing. The proposed CNN model gave an average accuracy of 0.886 and 0.884 for 16 bits and 8 bits, respectively. The corresponding false positive rates are down to 7.00% and 8.29% for 16 bits and 8 bits, respectively. We have evaluated 8 different conventional ML classifiers. The second best results were achieved by AB, which have a slightly lower detection accuracy than CNN. The AB gave an average accuracy of 0.80 and 0.793, where the corresponding false positive rates are 15.43% and 16.00% for 16 bits and 8 bits, respectively.

It can be concluded that for deep learning CNN, the average accuracy increase is 10.75% and 11.48% with lower false positive rates compared to Adaboost for 16 bits and 8 bits, respectively. As per our analysis on both 16-bit and 8-bit, 16-bit outperforms 8-bit by a tiny margin. Overall CNN performs better compared to conventional ML classifiers, but ML classifiers required less computational resources.

In the future, it might be conceivable to incorporate the region extraction into the CNN model as it at its core is an object detection problem. However, this will require additional testing and more data to conclude.

REFERENCES

[1] A. Poredos and A. Kitanovski, “District heating and cooling for efficient energy supply,” 2011 International Conference on Electrical and Control Engineering, pp. 5238-5241, Sep. 2011.

[2] M. Olsson, “Long-term thermal performance of polyurethane insulated district heating pipes,” Chalmers University of Technology, 2001 (PhD thesis).

[3] M. Fröling, “Environmental and thermal performance of district heating pipes,” Chalmers University of Technology, 2001 (PhD thesis).

[4] B Bøhm and M Borgström, “A comparison of different methods for in-situ determination of heat losses from district heating pipes,” *Technical University of Denmark*, 1996.

[5] J. Ahlberg O. Friman, P. Follo and S. Sjökvist, “Methods for large-scale monitoring of district heating systems using airborne thermography,” IEEE Transaction on Geoscience and Remote Sensing, vol. 52, pp. 5175-5182, 2014.

[6] A. Berg, J. Ahlberg, and M. Felsberg, “Enhanced analysis of thermographic images for monitoring of district heat pipe networks,” Pattern Recognition Letters 83, vol. 83, pp. 215-223, 2016.

[7] A. Berg and J. Ahlberg, “Classification of leakage detections acquired by airborne thermography of district heating networks,” 2014 8th IAPR Workshop on Pattern Reconition in Remote Sensing, pp. 1-4, Aug 2014.

[8] Sergiu Deitsch, Vincent Christlein, Stephan Berger, Claudia Buerhop-Lutz, Andreas K. Maier, Florian Gallwitz, and Christian Riess, “Automatic classification of defective photovoltaic module cells in electroluminescence images,” *CoRR*, vol. abs/1807.02894, 2018 (on review).

[9] MNIST database of handwritten digits: http://yann.lecun.com/exdb/mnist/.

[10] Hierarchical Data Format 5 (HDF5): https://support.hdfgroup.org/HDF5/doc/index.html.

[11] Yan and Xin, Linear Regression Analysis: Theory and Computing, World Scientific, 2009, pp. 1-2.

[12] R. O. Duda, P. E. Hart, and D. G. Stork, Pattern Classication, 2nd ed. John Wiley & Sons, 2001.

[13] T. Hastie a R Tibshirani and J. Friedman, The elements of statistical learning, 2nd ed. Springer, 2008.

[14] Webb and Geoffrey I., Naïve Bayes, In: Sammut C., Webb G.I. (eds), Springer US, Boston, MA, 2010,.

[15] Mucherino, Antonio, Papajorgji, Petraq J., Pardalos, and Panos M., k-Nearest Neighbor Classification, Springer, New York, 2009, vol. 34, pp. 83–106.

[16] Graham Williams, Decision Trees, Springer New York, 2011, pp. 205-244.

[17] L. Breiman, L. Machine Learning, 2001, pp. 5-32.

[18] C. M. Bishop, Pattern Recognition and Machine Learning, Springer, 2006.

[19] Claude Sammut and Geoffrey I. Webb, Eds., *Decision Stump*, pp. 262–263, Springer US, Boston, MA, 2010.

[20] D. P. Kingma and J. Lei Ba, “Adam : A method for stochastic optimization,” In 3rd International Conference for Learning Representations, pp. 1-15, 2015.

[21] Keras library : https://keras.io.