Semi-supervised rail defect detection from imbalanced image data

Hajizadeh, Siamak; Nunez Vicencio, Alfredo; Tax, David

Publication date
2016

Document Version
Accepted author manuscript

Published in
Proceedings of the 14th IFAC Symposium on Control in Transportation Systems, CTS 2016, Istanbul, Turkey

Citation (APA)
Hajizadeh, S., Nunez Vicencio, A., & Tax, D. (2016). Semi-supervised rail defect detection from imbalanced image data. In T. Acarman (Ed.), *Proceedings of the 14th IFAC Symposium on Control in Transportation Systems, CTS 2016, Istanbul, Turkey* (pp. 1-6)

Important note
To cite this publication, please use the final published version (if applicable).
Please check the document version above.

Please cite as: S. Hajizadeh, A. Núñez, and D. Tax, "Semi-supervised rail defect detection from imbalanced image data".
Proceedings of the 14th IFAC Symposium on Control in Transportation Systems, CTS 2016, Istanbul, Turkey, 18-20 May, 2016.

Semi-supervised Rail Defect Detection from Imbalanced Image Data

Siamak Hajizadeh, * **Alfredo Núñez,** * **David M.J. Tax** *

* *Delft University of Technology*
Stevinweg 1, 2628 CN, Delft, the Netherlands
e-mail: {S.Hajizadeh, A.A.Nunezvicencio, D.M.J.Tax}@tudelft.nl.

Abstract: Rail defect detection by video cameras has recently gained much attention in both academia and industry. Rail image data has two properties. It is highly imbalanced towards the non-defective class and it has a large number of unlabeled data samples available for semi-supervised learning techniques. In this paper we investigate if positive defective candidates selected from the unlabeled data can help improve the balance between the two classes and gain performance on detecting a specific type of defects called Squats. We compare data sampling techniques as well and conclude that the semi-supervised techniques are a reasonable alternative for improving performance on applications such as rail track Squat detection from image data.

Keywords: imbalance data, semi-supervised learning, rail image data, rail defect detection

1. INTRODUCTION

The rail network in the Netherlands has over 6000 kilometers of rail track. The tracks are worn due mostly to the effect of train traffic and the interaction with small defects that start to grow on various components. One specific type of defect called "Squat", forms as the result of repeated impacts caused by the running of vehicle wheels over smaller trivial defects and other rail components such as welds, switches and joints (Molodova et al. (2014)). If the formed squats are not detected and treated on time (e.g. by rail grinding) they grow further resulting higher maintenance costs. Among several automated inspection systems, visual inspection using video cameras has become more popular within the past few years: Li and Ren (2012); De Ruvo et al. (2008); Mandriota et al. (2004), mostly because of its simplicity and accessibility. With other inspections systems, such as physical measurements (e.g. in Molodova et al. (2014); Thomas et al. (2007)) the ground truth source of labeling the data is often provided via another type of measurement that needs to be collected separately. Images however can be manually labeled by human experts without the need for another validation source.

Our initial goal is to test the applicability of image data for detection of rail surface defects. Rail images data (and also other rail datasets) has a special characteristic by its nature. The dataset (normally from a well-maintained track) includes a large majority class (the non-defective rail) together with a much smaller minority class of defects. On Dutch passenger transport networks, often less than 1 percent of the inspected track length has some classes of defects on it. Machine learning literature has several methods to compensate for the imbalance of the classes (See He and Garcia (2009) for a review of these methods). One general direction studies the application of different sampling strategies (Chawla (2005)) to restore the balance with often notable improvements.

One other feature of the rail inspection data is that often a large number of unlabelled samples is available. The rail image data comes from a measurement train equipped with a high frame rate camera that covers hundreds of kilometers of rail every month. Labeling this data is done manually and therefore is time consuming. This leaves a large proportion of the data unlabeled. This is particularly interesting here because the unlabelled samples might help improve the data imbalance. Besides sampling, an option is to try to find potential positive (defective) samples from the unlabeled set and to add them to the training data. Several learning techniques have been proposed in the domain of *semi-supervised learning* to exploit information from the unlabelled data to the advantage of classification accuracy (Chapelle et al. (2006)). Therefore in this paper, we try a new application of such techniques that is to use semi-supervised learning for finding new candidate samples to extend the minority class. We would like to know if such techniques are comparable to the sampling techniques, that are proven to result in increased performance for almost all imbalanced data applications (Van Hulse et al. (2007)).

As we will discuss later in our experiment results, already with random under-sampling of the normal class, detection results are improved. This indicates that the imbalance between the two classes is indeed a problem. Therefore, another goal of this paper is to experimentally investigate how unlabeled examples can benefit defect detection from image data. We compare a number of well-known semi-supervised learning techniques (self-training, co-training, TSVMs, label propagation and weighted samples) against and also in combination with sampling approaches such as random under- and over-sampling and Synthetic Minority Oversampling Technique (SMOTE), to achieve the best output. The rest of the paper is as follows. In Section

* STW/NWO, ProRail

Please cite as: S. Hajizadeh, A. Núñez, and D. Tax, "Semi-supervised rail defect detection from imbalanced image data".
Proceedings of the 14th IFAC Symposium on Control in Transportation Systems, CTS 2016, Istanbul, Turkey, 18-20 May, 2016.

2 we describe the problem in details and shortly discuss the applied techniques. Section 3 presents data processing and techniques used for extracting features from our rail image data. Section 4 discusses our experimental design, our findings and we finalize with conclusions.

2. PROBLEM AND METHODS

We are interested in classifying the rail image data into two classes. One class is all variations of Squats. The other class contains other rail segments that mostly consists of normal healthy rail, but also includes welds, insulated joints, benign defects, Squat seeds and potential other defects such as small cracks. The overwhelming majority of the objects in our dataset belong to the non-Squat rail objects. For convenience we will call this objects the normal class. Out of the 21979 labelled cases, 21412 are normal rail objects. The remaining 567 samples form our *target* Squat class that are less than 3% of the rail image dataset. The consequence of this severe imbalance is that the minority class is often suppressed when training a classification model resulting high false negative rate (Squats that are classified as normal) in performance evaluation. In addition, the fact that the coloring and texture of the healthy normal rail (see Figure 2) can vary drastically from one example to another causing a high class variance, makes the effect of data imbalance more severe. An indication of this issue is observed by random under-sampling of the normal class. In case of our data, random under-sampling the majority class increases classification accuracy under almost all settings (See Section 4).

Squats are a type of late-stage defects that grow from smaller defects (often referred to as Squat seeds) on top rail surface (Grassie (2012)). Any small indentation, damage, or structure anomaly that is in direct impact with the wheels of the rolling stock, can act as a Squat seed and is prune to further growth to becoming a Squat. Not all small rail defects are seeds for Squats though. On rail, there are many smaller spots (in terms of depth or length) that are often benign and disappear by wear. Figure 1 shows examples of Squats that range from smaller to fully grown Squats. In Figure 2 a number of smaller defects and healthy rail samples are illustrated. There are various types of Squats depending on their sizes and shapes. There are 3 classical types: A, B, and C, with A being the smallest hence least severe and C the severest and often the largest in both length and depth. For automated classification however, we consider all Squats as one class (the target Squat class) since in real world data, these classes have smooth transitions and lack clear boundaries. The Squat seeds and other benign defects [1] on rail surface are all left out of the Squat class as objects of normal class in our dataset. Here we use a number of these commonly applied SSL techniques such as self-training and co-training to identify and add new positive examples to our training dataset in order to improve the class balance. We compare their performance against few sampling techniques such as random under-sampling of majority class, Gaussian noise over-sampling of minority class and SMOTE, that are known to be effective for learning from imbalance data. The sampling techniques are often used to improve the balance between the classes, but are mostly applied to fully supervised cases. In our case we will test the sampling techniques while discarding the unlabeled data. We also test the effect of combining under-sampling of the majority class with the SSL techniques to see if it leads to higher performance. Rest of this section describes the tested techniques in short.

[1] The question whether or not a small defect is benign or is sufficiently critical to grow to a Squat does not have a rigid Yes or No answer by their sight. Depending on their size, material structure of their location, and the type of wheel-rail contact forces these defect can grow to a Squat.

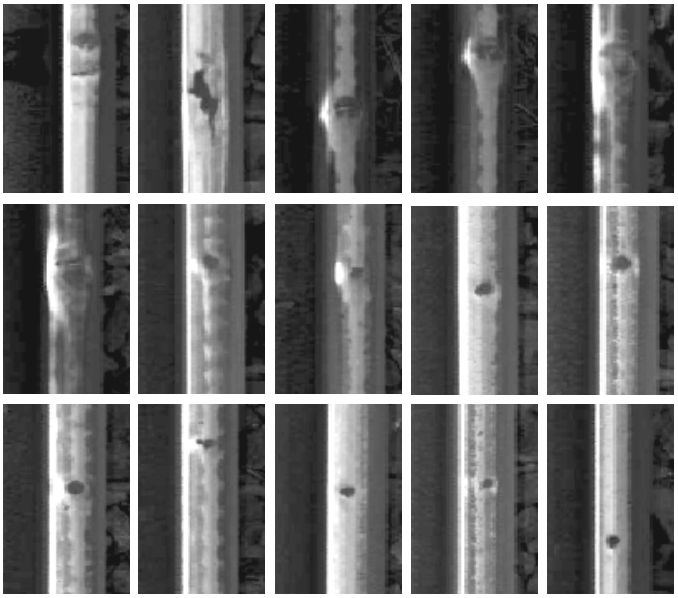

Fig. 1. A number of samples from the Squat class in rail video data.

Self training: Self training is commonly the simplest semi-supervised technique to exploit information from unlabeled examples. In self training, the output score of a classifier on unlabeled examples, is used to select new unlabeled objects and to label them as predicted by the classifier. The new objects are then added to the training set and the classifier is re-trained, assuming that the newly added training objects will help improve the performance. Although there are several variations, in its most common form objects with the highest confidence level (based on the output score) are selected at each iteration and added to the training set. This process continues for an specific number of maximum iterations or until satisfaction of a convergence criterion. In our experiments, we use two types of self-training. Aside from the process described above, we test the approach introduced in Elkan and Noto (2008) for learning from only positive and unlabeled data. In this case the newly added objects are weighted by their posterior probabilities that are acquired translating the classifier output score by a sigmoid fit called Platt scaling, Platt (1999). In both cases, we adapt the method to only add new positive candidate objects to the training set, since our goal is to improve the balance between the minority and the majority classes.

Co-training: Co-training (Blum and Mitchell (1998)) is similar to self training in that it also uses classifier outputs to choose new training candidates from unlabeled data. As the name suggests, co-training uses two classifiers that are trained using two *separate* feature sets. Co-training

Please cite as: S. Hajizadeh, A. Núñez, and D. Tax, "Semi-supervised rail defect detection from imbalanced image data".
Proceedings of the 14th IFAC Symposium on Control in Transportation Systems, CTS 2016, Istanbul, Turkey, 18-20 May, 2016.

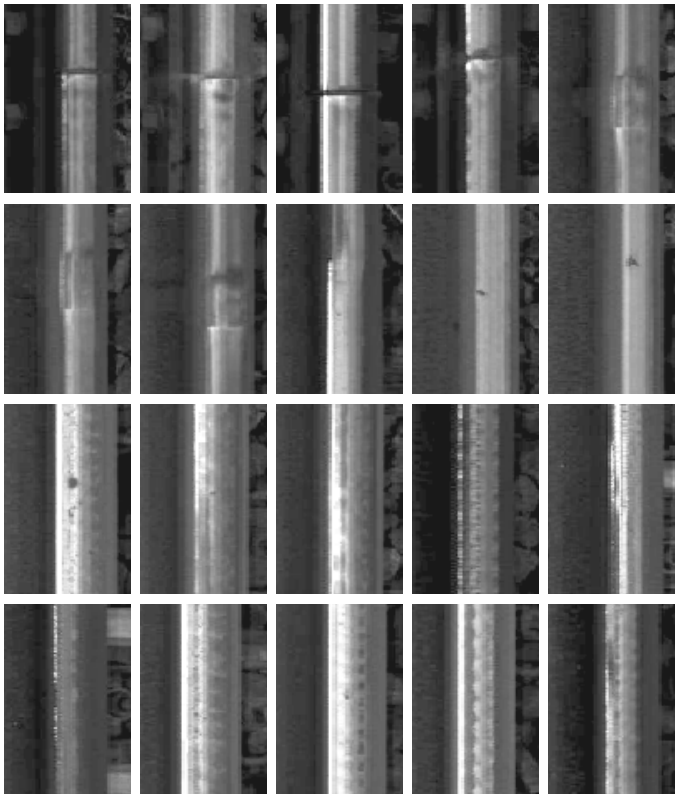

Fig. 2. Samples from the normal class include healthy rail, small spots, welds, insulated joints, etc.

assumes that the two separate feature sets have mutually independent distributions, conditional on the given class labels. It has been shown empirically thought that even if this criterion does not hold, co-training can help improve classification performance (Balcan et al. (2004)). Often in vision based applications of co-training, gradient based features such as HoG, SIFT, spacial pyramids have been used together with features built using basis functions such as Gabor filters. We understood in our initial tests that gradient based features do not result in good classification due to a number of reasons such as high variability of the shapes and low resolution and high noise of the rail images. Gabor and DCT features both fairly represented the two classes in the initial tests where individual classifiers were trained on each feature set producing above 90 % accuracies [2]. Therefore although they do not satisfy the independence criterion, our experiments show that using Gabor features and DCT feature for co-training makes steady improvement of prediction accuracy. Similar to the case of self-training above, we also try co-training with weighted objects.

Transductive SVMs: *Transductive* Support Vector Machines are a modification of supervised SVMs that are able to learn from unlabeled data distribution. The original idea of the SVMs is to find a linear boundary that optimize the separation between the two classes. TSVMs (also referred to as semi-supervised support vector machines or S^3VMs for short) approximate lowest error separation while adjusting new labels for the unlabeled data. This interprets in theory to choosing a linear boundary that passes through low density regions of the data space (Zhu (2005)). The term transductive is meant to convey that the model fits only to describe the training data and does not have predictive capabilities for unseen examples. However in several applications, TSVMs have been used for *inductive* prediction of unseen data. We also use TSVM for induction. Among the several variation, we chose the *svmlin* implementation with optimizations by Sindhwani and Keerthi (2006).

Label propagation: Label propagation assigns labels to the unlabeled data iteratively. Initially all labeled and unlabeled data points are represented in terms of k neighboring data points with optimized reconstruction weights. These weights are then used to create a graph representation. At each iteration, by using the neighborhood graph, a proportion of the label information is *absorbed* by the unlabeled data and the process continues until convergence, Wang and Zhang (2008). Label propagation is therefore transductive in nature, but the propagation mechanism can be used for induction. In our case, we use label propagation to identify only new positive data and we use them for re-training our classifier, to be consistent with the other tested techniques.

Random majority undersampling: The simplest way to shrinking the majority class in order to reduce class imbalance is to make a random selection of its objects and discard the rest. Our results show that random under sampling is in fact effective in gaining performance. We try random under sampling of the majority also in combination with other techniques discussed here, to see if we can get additional advantage in performance.

Minority oversampling with noise: The minority class is oversampled by randomly duplicating the available minority objects and optionally adding a small ratio of Gaussian noise. This technique is also common as one of the most basic yet sometimes effective techniques.

SMOTE: Synthetic Minority Oversampling Technique is a technique to oversample a class by extrapolating new synthetic data points on the line connecting a class object and one of its randomly selected nearest neighbors (Chawla et al. (2002)). The procedure continues until the required number of synthetic data points are generated. SMOTE has been proven to enhance performance significantly in many domains. A number of variations of SMOTE have also been introduced, e.g. in Han et al. (2005), but we test only the original algorithm and compare it to semi-supervised techniques that populate minority class by selecting new samples from the unlabeled data.

3. DATA AND FEATURES

Our data comes from a high frame rate camera that is mounted on a measurement vehicle. It consists of 37 high frame rate videos covering approximately 350 kilometers of track, or 700 kilometers of rail. The camera captures the rail from the top view. To build the dataset, initially each video frame is processed by segmentation to extract the rail top segment from the surrounding background. Right rail images are flipped along the vertical axis in order to

[2] We tried both radial basis support vector machines (SVM) and feed-forward back propagation neural networks with number of hidden nodes twice the size of features.

Please cite as: S. Hajizadeh, A. Núñez, and D. Tax, "Semi-supervised rail defect detection from imbalanced image data".
Proceedings of the 14th IFAC Symposium on Control in Transportation Systems, CTS 2016, Istanbul, Turkey, 18-20 May, 2016.

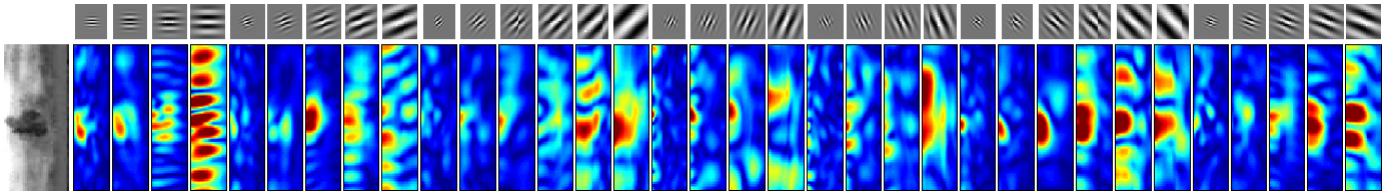

Fig. 3. An example rail segmented object, is convoluted with an array of different parameterization of Gabor function family.

make left and right rails as indifferent as possible. In total 21979 objects are labeled manually from 7 selected videos. The other 30 videos are used to extract 718520 unlabeled objects that include all types Squat and non-Squat objects. The data is only partially labeled since the manual labeling process needs expert domain knowledge and takes a lot of time. Finally two sets of features are extracted from each rail segment.

The first feature set is the result of the convolution of a segment with a set of parameterizations of Gabor function family which is a common feature extraction method in image processing (Manjunath and Ma (1996)). Different parameter values correspond to different wavelengths and rotations of the Gabor function. Figure 3 illustrates an example of an extracted rail segment that has been convoluted with the applied stack of Gabor function family of varying rotation and size parameters. We should note that we omit the function parameter values that correspond to vertical alignment of the wavelets. The reason is that most marks and texture structures on rail surface are aligned with the longitudinal (vertical in the photos) direction due to the structure of the rail. Vertical wavelets excite almost the entire rail segment which makes them irrelevant for the defect detection from low resolution images.

Using the filter results, we are also able to locate the object of interest along the rail segment. Instead of using the entire segment as one object, we use the excitation magnitude values to locate the point of interest along the rail and cut out the located object. Figure 4 shows the mostly excited point located by summing filter amplitudes over short side of all filter results. The feature values are acquired by averaging the excitation amplitude at the neighborhood of the defect on the rail segment.

The second feature set is constructed using the 2 dimensional discrete cosine transform (DCT). To get the final features, we decrease the resolution of the transform result by averaging ranges of frequencies. We consider higher cosine frequencies less important for identifying the defects because very high frequency components are often due to the noise in the images or occasionally due to insignificant small spots on rail. Therefore instead of treating each frequency component $X(u, v)$ as equal, we average

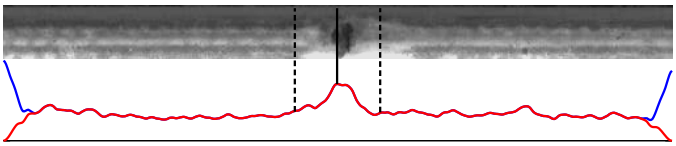

Fig. 4. Object of interest cut out by summing the filter amplitudes along the short segment side. The edges are trimmed to remove the leakage.

$X(U_i, V_j)$ where $U_i = \{u \in \mathbb{N} \mid 2^{i-1} \leq u < 2^i\}$ and $V_j = \{v \in \mathbb{N} \mid 2^{j-1} \leq v < 2^j\}$ for all non-negative integers i and j. Figure 5 shows how this averaging is projected on the transform output to calculate the final features.

The advantage of DCT is that it can compresses most the content information of the images into the first few modes of the transform outcome. This property of DCT has been used in image compression techniques. We discard phase information by taking the absolute value and average over the magnitudes.

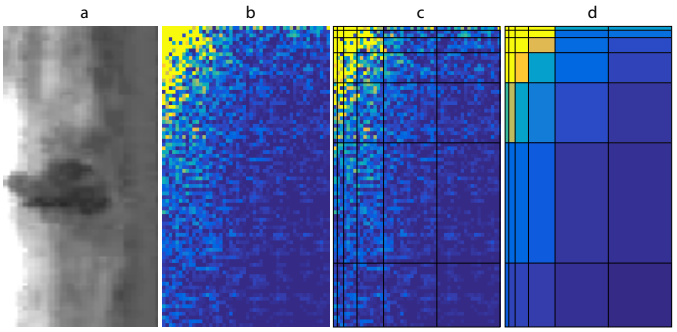

Fig. 5. Discrete Cosine Transform (b) of an extracted rail segment featuring a Squat (a). The resolution of the transform outcome is reduced by taking the average using a nonlinear grid (c), to build the final data features (d).

4. EXPERIMENTAL RESULTS

At each round we randomly choose half of each class for training and leave the other half for testing. Initially 3 classifiers are trained using the Gabor features, DCT features, and all features. We also combine the output of Gabor and DCT classifiers in a soft-voting style by averaging their posterior probability on the test data as a fourth classifier. The trained classifiers are then used for the semi-supervised and sampling techniques discussed in Section 2. Each of the techniques: self-training (SR) and self-training with weights (SRW), co-training (CT) and co-training with weights (CTW), transductive SVM (TSVM), nearest neighbor linear label propagation (LP), random Gaussian noise minority oversampling (RMO) and SMOTE, are once applied to the original training data (labeled training data and unlabeled samples) and once after random majority undersampling (RMU) is applied to the training set. New positive candidates that are selected from the unlabeled are added to the training set and the classifiers are re-trained [3] and re-tested on the same

[3] This is with the exception of TSVM where the training data is augmented with the whole unlabeled objects that receive new label 0 to set them apart from the labeled data.

Please cite as: S. Hajizadeh, A. Núñez, and D. Tax, "Semi-supervised rail defect detection from imbalanced image data".
Proceedings of the 14th IFAC Symposium on Control in Transportation Systems, CTS 2016, Istanbul, Turkey, 18-20 May, 2016.

Table 1. F1-score, AUC, and Accuracy averaged for 80 folds of tests

			Base	ST/CT	STW/CTW	LP	TSVM	RMO	SMOTE	Cheating
F1 score	Original	Gabor	0.7082	0.7825	0.7723	0.7715	0.7759	0.8018	0.7944	-
		DCT	0.7326	0.7849	0.7779	0.7968	0.7651	0.7987	0.7997	-
		G. + D.	0.7579	0.8134	0.8060	**0.8341**	0.8161	0.8209	0.8111	-
		Soft vote	0.7270	0.7771	0.7700	0.7909	0.8059	0.8201	0.8173	-
		Co-train	-	**0.8575**	**0.8533**	-	-	-	-	-
	RMU	Gabor	0.8127	0.8665	0.8642	0.8210	0.8340	0.8524	0.8462	0.8727
		DCT	0.8185	0.8512	0.8488	0.8362	0.8299	0.8510	0.8437	0.8647
		G. + D.	0.8444	**0.8769**	**0.8742**	0.8613	0.8607	0.8687	0.8609	0.8925
		Soft vote	0.8372	0.8695	0.8673	0.8434	0.8533	**0.8746**	0.8702	0.8813
		Co-train	-	0.8568	0.8595	-	-	-	-	-
AUC	Original	Gabor	0.9603	0.9675	0.9638	0.9618	0.9740	0.9762	0.9769	-
		DCT	0.9624	0.9716	0.9668	0.9669	0.9698	0.9762	0.9772	-
		G. + D.	0.9742	0.9787	0.9771	0.9771	0.9779	0.9824	0.9824	-
		Soft vote	0.9754	0.9774	0.9763	0.9758	0.9794	0.9830	0.9830	-
		Co-train	-	0.9783	0.9776	-	-	-	-	-
	RMU	Gabor	0.9743	0.9756	0.9750	0.9699	0.9740	0.9782	0.9771	0.9786
		DCT	0.9751	0.9758	0.9752	0.9726	0.9700	0.9786	0.9775	0.9795
		G. + D.	0.9808	0.9813	0.9811	0.9788	0.9778	0.9831	0.9825	0.9849
		Soft vote	0.9813	0.9812	0.9812	0.9785	0.9796	0.9833	0.9827	0.9834
		Co-train	-	0.9791	0.9798	-	-	-	-	-
Accuracy	Original	Gabor	0.8986	0.9180	0.9152	0.9152	0.9158	0.9239	0.9212	-
		DCT	0.9053	0.9186	0.9169	0.9223	0.9129	0.9236	0.9236	-
		G. + D.	0.9124	0.9279	0.9257	0.9346	0.9287	0.9308	0.9274	-
		Soft vote	0.9040	0.9177	0.9157	0.9214	0.9251	0.9306	0.9295	-
		Co-train	-	0.9348	0.9343	-	-	-	-	-
	RMU	Gabor	0.9269	0.9398	0.9394	0.9282	0.9311	0.9380	0.9356	0.9403
		DCT	0.9290	0.9359	0.9356	0.9325	0.9300	0.9384	0.9354	0.9377
		G. + D.	0.9379	0.9447	0.9440	0.9420	0.9416	0.9454	0.9426	0.9526
		Soft vote	0.9356	0.9427	0.9423	0.9361	0.9387	0.9474	0.9457	0.9451
		Co-train	-	0.9386	0.9400	-	-	-	-	-

testing data. Finally we also train and evaluate what we call a *cheating* classifier, to find the upper limit of performance gain from unlabeled examples. To train the cheating classifier, first a classifier is trained on all the labeled data and is used for selecting positive candidates. Then the selected unlabeled samples are added to the training set and a new *cheating* classifier is trained. The positive candidates are therefore chosen with information from the testing data as well, hence the classifier would be cheating.

We run 80 rounds of cross validation. For all tested techniques, where we select from unlabeled pool or add generated data points, a constant number of 2500 positives are added to the positive class. When undersampling, out of the 10706 normal training objects, half are randomly removed. We train SVM classifiers for all our tests using LIBSVM by Chang and Lin (2011) and svmlin libraries. To measure the classification performance, 3 common metrics are calculated after each test: Accuracy, Area under the ROC curve (AUC), and F1 score. The results are then averaged over the 80 cross validation rounds. Accuracy is the most common measure of performance. It is simply the ratio of the correct predictions in all the tested data regardless of class sizes and possible imbalance. Therefore a large majority class can easily overrule the loss of accuracy made by false prediction made on the minority class. AUC is less sensitive to data imbalance since it is independent of a threshold and direct predictions. Instead, the degree to which the positive and negative test samples are separated determines its outcome. However AUC has a number of shortcomings as well (Hand (2009)). One is that an evaluation of performance without a classification threshold can be a questionable choice. In a real-world application one needs a threshold to predict outcomes, while integrating over a range of different thresholds in AUC can interpret to summing over different misclassification costs (Ferri et al. (2011)), which is unrealistic for the rail defect detection.

F1 score can be a suitable choice when one is interested in evaluating performance towards detection of one target class. For the Squat class for example, we are interested to know the ratio of the number of correct Squat detections over both the total number of tested Squats and over the total number of detections, respectively yielding *recall* and *precision*. The F1 score that is the harmonic mean of recall and precision, conveys these information and thus is a suitable choice for performance metric. Although it is still affected by data imbalance, we trust F1 score to suit more our idea of performance in the rail defect detection. Table 1 shows evaluations of AUC, F1 score and accuracy for all the tested techniques. The top 3 are in bold for both cases of original training set and the random majority undersampling (RMU).

The results show that training on both of the feature sets Gabor and DCT always yields better performance than discarding one. Also RMU improves performance generally with only few exceptions. In the case where RMU is applied to the training set, self-training with all the features shows the highest improvement of performance on both feature sets and on their combinations. All other semi-supervised and sampling techniques show comparable of performance gains over the base setting. When the original non-reduced data is used for training, co-training

shows remarkable improvement of the performance. In both cases using sample weights seem to loose overall performance compared to the original self- and co-training techniques.

5. CONCLUSIONS

Class imbalance is a significant problem for automated classification in rail data. Naive random oversampling of the minority does not add information to the training set, yet by merely improving the size balance yields significant performance improvement for most of the tested cases. Reasonable detection results show that rail image data is a valuable source of information for automated processing and detection. Here we tested SSL techniques in comparison with different samplings of the classes. From the experiments we conclude that reduction of imbalance using the unlabeled data is a proper alternative to sampling.

The cheating classifier in results section, gives us an indication of how far better the performance can get by improving the selection of candidate unlabeled positive examples. Our goal here is to prove applicability of semi-supervised techniques for data imbalance treatment, and not fine tuning the techniques for maximal performance. However the cheating classifier performances show that it might be still possible to reach higher accuracies using the unlabeled data in future extension of the work.

ACKNOWLEDGEMENTS

This research is supported by ProRail and the Dutch Technology Foundation STW, which is part of the Netherlands Organization for Scientific Research (NWO), and which is partly funded by the Ministry of Economic Affairs.

REFERENCES

Balcan, M.F., Blum, A., and Yang, K. (2004). Co-training and expansion: Towards bridging theory and practice. In *Advances in neural information processing systems*, 89–96.

Blum, A. and Mitchell, T. (1998). Combining labeled and unlabeled data with co-training. In *Proceedings of the eleventh annual conference on Computational learning theory*, 92–100. ACM.

Chang, C.C. and Lin, C.J. (2011). LIBSVM: A library for support vector machines. *ACM Transactions on Intelligent Systems and Technology*, 2, 27:1–27:27.

Chapelle, O., Schölkopf, B., and Zien, A. (2006). *Semi-supervised learning*. MIT press Cambridge.

Chawla, N.V. (2005). Data mining for imbalanced datasets: An overview. In *Data mining and knowledge discovery handbook*, 853–867. Springer.

Chawla, N.V., Bowyer, K.W., Hall, L.O., and Kegelmeyer, W.P. (2002). SMOTE: synthetic minority over-sampling technique. *Journal of artificial intelligence research*, 16, 321–357.

De Ruvo, P., De Ruvo, G., Distante, A., Nitti, M., Stella, E., and Marino, F. (2008). A visual inspection system for rail detection; tracking in real time railway maintenance. *Open Cybernetics & Systemics Journal*, 2, 57–67.

Elkan, C. and Noto, K. (2008). Learning classifiers from only positive and unlabeled data. In *Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining*, 213–220. ACM.

Ferri, C., Hernández-Orallo, J., and Flach, P.A. (2011). A coherent interpretation of AUC as a measure of aggregated classification performance. In *Proceedings of the 28th International Conference on Machine Learning*, 657–664.

Grassie, S. (2012). Squats and squat-type defects in rails: the understanding to date. *Proceedings of the Institution of Mechanical Engineers, Part F: Journal of Rail and Rapid Transit*, 226(3), 235–242.

Han, H., Wang, W.Y., and Mao, B.H. (2005). Borderline-SMOTE: a new over-sampling method in imbalanced data sets learning. In *Advances in intelligent computing*, 878–887. Springer.

Hand, D.J. (2009). Measuring classifier performance: a coherent alternative to the area under the ROC curve. *Machine learning*, 77(1), 103–123.

He, H. and Garcia, E. (2009). Learning from imbalanced data. *IEEE Transactions on Knowledge and Data Engineering*, 21(9), 1263–1284.

Li, Q. and Ren, S. (2012). A real-time visual inspection system for discrete surface defects of rail heads. *IEEE Transactions on Instrumentation and Measurement*, 61(8), 2189–2199.

Mandriota, C., Nitti, M., Ancona, N., Stella, E., and Distante, A. (2004). Filter-based feature selection for rail defect detection. *Machine Vision and Applications*, 15(4), 179–185.

Manjunath, B.S. and Ma, W.Y. (1996). Texture features for browsing and retrieval of image data. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 18(8), 837–842.

Molodova, M., Li, Z., Nunez, A., and Dollevoet, R. (2014). Automatic detection of squats in railway infrastructure. *IEEE Transactions on Intelligent Transportation Systems*, 15(5), 1980–1990.

Platt, J. (1999). Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods. *Advances in Large Margin Classifiers*, 10(3), 61–74.

Sindhwani, V. and Keerthi, S.S. (2006). Large scale semi-supervised linear SVMs. In *Proceedings of the 29th annual international ACM SIGIR conference on Research and development in information retrieval*, 477–484. ACM.

Thomas, H., Heckel, T., and Hanspach, G. (2007). Advantage of a combined ultrasonic and eddy current examination for railway inspection trains. *Insight-Non-Destructive Testing and Condition Monitoring*, 49(6), 341–344.

Van Hulse, J., Khoshgoftaar, T.M., and Napolitano, A. (2007). Experimental perspectives on learning from imbalanced data. In *Proceedings of the 24th international conference on Machine learning*, 935–942. ACM.

Wang, F. and Zhang, C. (2008). Label propagation through linear neighborhoods. *IEEE Transactions on Knowledge and Data Engineering*, 20(1), 55–67.

Zhu, X. (2005). Semi-supervised learning literature survey. *world*, 10, 10.