
European Study Group with Industry 121
Danmarks Meteorologisk Institut. Fog detection

Mark Lyksborg · Monica J. Emerson ·
David Norsk · Kamil Kulesza · Henrik G.
Petersen · Christian Henriksen

January 25, 2018

Abstract In this paper a framework to detect the presence of fog in optical images is presented. The data was provided by Danmarks Meteorologisk Institut (DMI) and it consisted of images from cameras at four different locations in Denmark. Labelling of the images as foggy or not foggy was performed by ourselves, image analysts and not weather experts. The approach presented in this paper requires the training of a classifier that takes decisions based on a number of image features. Most of these image features were previously used for haziness detection and are gradient based. The random forest classifier was able to predict the occurrence of fog in most of the cases. Misclassifications were mainly due to the categorisation of the training set by non-experts, sun flares, camera movements and others.

Keywords Fog detection · Image features · Random Forest · Sun Streaks · Laplacian · Dark channel Prior · Sobel Filter

M. Lyksborg, M. J. Emerson, D. Norsk, C. Henriksen
DTU E-mail: mlyk@dtu.dk, monj@dtu.dk, dnor@dtu.dk, chrh@dtu.dk

K. Kulesza
University of Warsaw E-mail: Kamil.Kulesza@damtp.cam.ac.uk

H. G. Petersen,
SDU E-mail: hgp@mmtmi.sdu.dk

1 Introduction

Decreased visibility in the form of fog, snow, rain, darkness, sunset or sunrise is one of the major causes of traffic accidents. For this reason, it is of great practical value to detect decreased visibility and warn drivers.

Various instruments that measure quantities related to reduced visibility are available in the market, such as haze meters or scatterometers. These are not extensively deployed in Denmark, however there is a network of traffic cameras scattered around the roads of the country. If the visibility wants to be calculated along the whole road network, it is cheaper to re-use the already deployed camera system.

Therefore, the aim of this project is to detect decreased visibility from optical images. Danmarks Meteorologisk Institut (DMI) provided data which contains examples of foggy and non foggy images. Therefore, the problem of decreased visibility is narrowed down to the distinction between these two scenarios. The reason for which fog causes decreased visibility is the presence of water particles in the air that create light scattering and absorption. For simplicity and reasons of time we limit our focus to day time imagery.

The prediction of fog occurrence can be used to provide information about the visibility conditions through broadcasting systems such as TV and radio; or automatic warnings on the screens placed over the roads. Moreover, it can improve numerical weather forecasting (NWC) which is not good at predicting fog, as it is a condition that takes place close to the ground. The results obtained here could be used to validate the estimates of NWC or fed into the models so as to improve future predictions.

In this paper, it is assumed that each camera is an individual system. Nevertheless, the whole network of cameras could be seen as a system. This will be further discussed in the Section 5.

Some of the challenges that this project presents are problem dependant and others are data dependant. Problem dependant challenges include varying lighting, weather conditions, cars moving, seasons, night time, shaken images and over exposure due to the sun shining directly into the camera. The latter will be mentioned in Section 2. Since this project has a limited time-frame our solution strategy was to look into literature and based on this knowledge investigate solutions proposed for solving the aforementioned challenges.

1.1 Literature study

Several papers deal with the problem of visibility detection or scoring from ordinary images. Some of these are more relevant than others, and some such as [7] are not useful. This method requires knowing a 3D model of the scene which is not feasible to create for all roads being monitored by traffic cameras since the estimation of such model requires, for example, multiple cameras looking at the same object/scene.

Another method which also requires several cameras was by [3], where fog is detected with the purpose of removing it from the images. The fog detection was done by estimating the entropy of stationary image regions under the assumption that image intensity entropy decreases with fog.

There are methods that are suitable for fog detection, even though they were created with another purpose. An example is [2], where an end-to-end denoise/deHaze model neural network model is trained for dehazing that could potentially also be used for fog detection. The method performs well compared to other methods but learning a good model requires a large amount of paired image data; each pair consisting of a clear view image and foggy image of the same scene. They used a large number of images ($n = 100000$) and required an efficient graphics card to learn the model, none of which we currently have. These requirements exclude its usage at current time but we believe it would lead to a superior fog detection method.

Another method which has been used for dehazing [5] is based on the dark channel prior which originates from a physics based modelling of the haze phenomena (further explained in Section 3.1). Many variations and refinements have been proposed since, such as in [8] and we shall later see it performs well for the purpose of fog detection.

A problem similar to fog detection is that of determining the focus of a camera. This problem has been addressed in the field of computer vision, area where many techniques for the calculation of focus-measures have been proposed [11]. These methods generally use the fact that image gradient magnitudes decrease as image blur increase. Therefore, methods extracting 1st [6], [4] and second order image information [9] are highly useful for producing a focus/fog measure and will be amongst the measures used to detect fog in this work.

The outline of this report is the following. In Section 2 the data will be presented, then the method for classification is described in Section 3. The results are presented in Section 4 and discussed in Section 5 where also some conclusions are drawn and ideas for future work are introduced.

2 Data

DMI provided 4781 images recorded at four different locations in Denmark: Aarhus, Esbjerg, Fanoe and Kastrup. As can be seen in Figure 1, the landscapes can vary quite substantially. The pictures were taken at set time intervals, which vary between the different cameras - 5, 10 or 15 minutes intervals. Along with the images, a list of days during which fog had occurred was supplied. Therefore, there was the need for manually annotating the individual images as foggy or no-foggy. This labelling was performed by the authors, which are not experts in meteorology. In Figures 2 and 3 example images for each case (no fog and fog) are shown.

Ideally, DMI would have liked to have the images classified with multiple labels following a scale from 0 to 3 where 0 indicates no visibility, 1 limited visibility, 2 moderate visibility and 3 full visibility. For a proof of concept, only a binary scale is used (fog, no fog) as the data had not been annotated by an expert and the visibility distance was not available.

Several data dependant challenges were identified during the initial data exploration stage, amongst these were:

- Time-stamp problems. The time-stamp for each image was at times randomly placed, such that a pitch black image was seen at 15:00 in the afternoon, sunrise at 3 am in the night, and sunset at 9 pm in July.



Fig. 1: Images from the four different cameras.

- Frozen pictures. In some cases the camera also seemed to have ”frozen” because the same picture was displayed for the entire day.
- Colour and, black and white mode. The cameras also seem to have some detection of light conditions where, if the amount of light falls below a certain threshold, the camera mode changes to black and white. This is problematic as some cameras are placed in an illuminated place, where a change to black and white never occurs, and others are lying in an area where there are no street lights.
- Lack of expert annotation. The fog/no fog annotation was performed in very rough manner by non experts. Knowing if fog had occurred on a specific day was not good enough to annotate accurately the images from the whole day.

The data was handled taking into account the problems mentioned above so the time-stamps were not used for the analysis, the frozen and night time pictures were removed and the annotation was performed for the individual images as accurately as possible.

3 Method

The data should be processed so as to obtain a classification that determines for each image whether it contains fog or not. For this purpose, a general framework with several steps is implemented (see Figure 4). First, a feature extraction step where different image features are calculated followed by some processing of these features such as normalisation. Then, our Random Forest classifier is trained with



Fig. 2: Thirty example images labelled as the no fog class

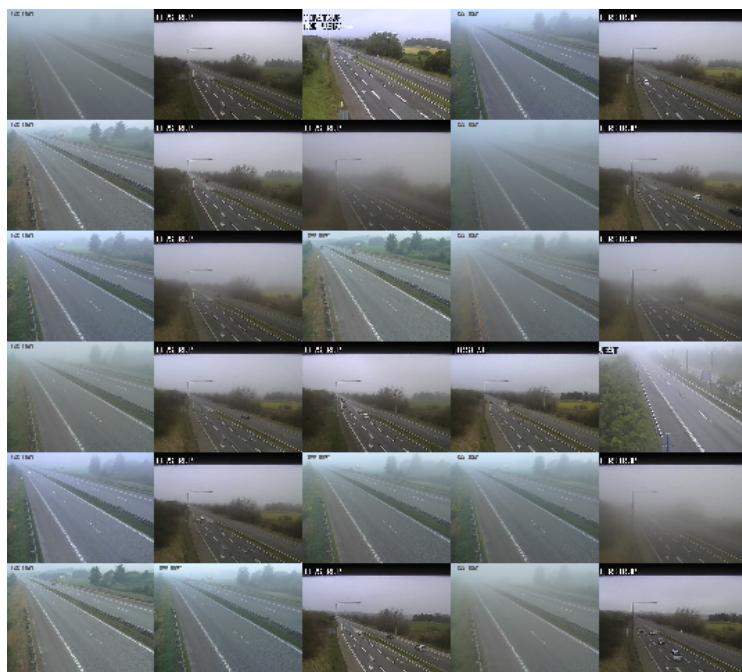


Fig. 3: Thirty example images labelled as the fog class

the image features of the annotated training set. However, there is the option of doing some post-processing of these features such as normalisation. Once the classifier has been trained, it is able to predict the class (fog or no fog) of un-seen images, i.e. images that do not belong to the annotated training set.

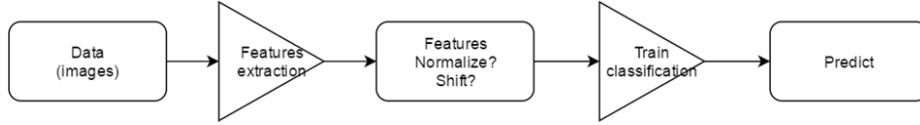


Fig. 4: Explains general setup for data handling, from input to final prediction.

3.1 Feature extraction

Several features were investigated and an exploratory analysis was performed on the images shown in Figure 5, which belong to the same road on the same day, one containing fog and the other one does not. All the features mentioned here were included in the framework.



Fig. 5: The same road on the same day with and without fog.

3.1.1 Sobel Edge filtering

Sobel edge filtering enhances strong changes in the intensity profile of an image. The analysis in [12] shows that edges far away in the image disappear if fog is present in the image, therefore Sobel filtering can be used for evaluating the presence of fog in an image. The filtering is performed in two stages, one for enhancing edges in the x-axis (filter G_x) and one for the y-axis (filter G_y). These two are defined as:

$$G_x = \begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix} \quad (1)$$

$$G_y = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix} \quad (2)$$

After filtering, the two resulting images are merged following Equation 3 so as to obtain the magnitude of the gradient, namely the Sobel filtered image, which contains the edges in all directions. The images obtained from applying the Sobel filtering technique to the images displayed in Figure 5 are shown in Figure 6.

$$S(m, n) = \sqrt{S_x(m, n)^2 + S_y(m, n)^2} \quad (3)$$

where S is the Sobel filtered image and, S_x and S_y are the convolution of the intensity image with the filters G_x and G_y respectively.

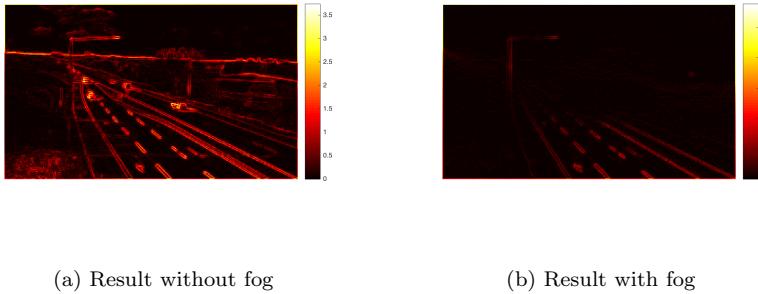


Fig. 6: Results from applying a Sobel filter to images of same road on the same day with and without fog.

As expected, the edges become less noticeable as the depth in the image increases. Two metrics could be used to calculate a visibility measure, these are described in [10]. The first one is based on calculating the variance of the Sobel filtered image S (see Equation 4) and the second is the Tenengrad focus measure (see Equation 5).

$$VAR_{SOB} = \sum_m^M \sum_n^N [S(m, n) - \bar{S}]^2 \quad (4)$$

$$TEN = \sum_m^M \sum_n^N [S(m, n)]^2 \quad (5)$$

3.1.2 Laplacian

The Laplacian filtering is based on the same ideas as the Sobel filtering in the sense that it also quantifies gradient related information. However, it encodes the second order information. Both first and second order derivatives carry information concerning blurriness (fog) levels.

The Laplacian filter used to generate the Laplacian of an image is shown in Equation 6, here in the case of a 3×3 matrix - but other sizes can be defined as well.

$$G = \frac{1}{6} \begin{bmatrix} 0 & -1 & 0 \\ -1 & 4 & -1 \\ 0 & -1 & 0 \end{bmatrix} \quad (6)$$

To get the Laplacian images, images such as the ones in Figure 5 are convolved with G resulting in Figure 7. Here a large difference between the two cases (fog and no fog) is visible.

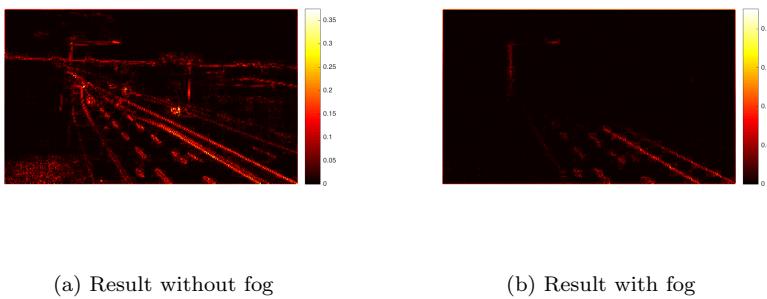


Fig. 7: Results from applying a laplacian filter to images of same road on the same day with and without fog.

Two different metrics can be used to sum up the information contained in a Laplacian filtered image L , both are described in [10]. The first is the sum of the absolute values of the pixels and the second is the variance of the absolute values of the pixels.

3.1.3 Dark Channel and Dark Channel Prior

The dark Channel Prior is an assumption that allows one to derive strong results about the presence of fog. He et al. in [5] used it to define an algorithm for dehazing an image. Many variations and refinements have been proposed since, see [8] for an overview and comparison of the methods.

The following equation provides a basic model of haze.

$$I(\mathbf{x}) = t(\mathbf{x})J(\mathbf{x}) + (1 - t(\mathbf{x}))A$$

where I is the image, \mathbf{x} is the position within image, J is the ideal image we would have without the presence of haze, and A is the airlight.

Furthermore, $t(\mathbf{x})$ is the transmission which is dependant on the depth and a transmission coefficient, which quantifies the presence of haze. The dark prior assumes that in every patch, neighbourhood of some fixed size around a pixel \mathbf{x} , there is a zero value in one of the channels (r, g or b) of a pixel. Using the dark

prior, and assuming that t can be considered constant in a patch, t and thus J can be reconstructed from I . Taking the minimum of the rgb channels of every pixel, calculated in a patch neighbourhood for each of the pixels \mathbf{x} , yields a pixel map D called the *dark channel*. Where no haze is present $t(\mathbf{x}) = 0$ then, according to the dark channel prior, $D(\mathbf{x}) = 0$. When haze has the maximal possible impact, $t(\mathbf{x}) = 1$ and according to the model, $D(\mathbf{x}) = \min A$, where the minimum is taken over the three values (rgb) of the airlight A . So the mean value of the dark channel should correlate positively with presence and density of the fog. This is confirmed by processing the two reference images for fog and no fog, results are shown in Figure 8.

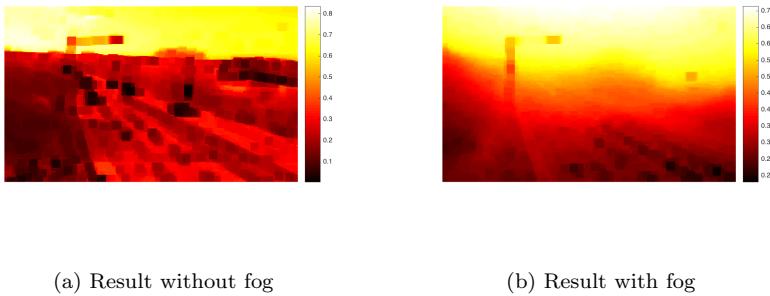


Fig. 8: Resulting dark channel for images of same road on the same day with and without fog.

The feature that can be derived from the dark channel is its centre of mass. As the distance from the camera to the objects in the upper part of the image is usually bigger than for those in the lower part, and the impact of haze increases with depth, the height coordinate of the centre of mass of the dark channel might say something about the presence of fog.

The 'light' channel $L(\mathbf{x})$ can be derived similarly as the dark channel, by maximising instead of minimising the definition. Then, the difference between light and dark channel, $L - D$, which is a measure of local contrast, is also used as a feature.

3.1.4 Overexposed pixels

Sometimes the sun was shinning directly onto the front lens leading to lens flares and images where parts of it were over exposed. An example of this is shown in Figure 12, (d). To be able to correctly classify such images, we added a feature for detecting overexposed pixels. A pixel is overexposed if one of the rgb-values is equal to the maximum possible (i.e 1.0, or 255 depending on image model). The feature will account for the fraction of overexposed pixels inside an image.

3.1.5 Time average heuristic for weighting robust edges

Images from the same camera contain edges that appear and disappear due to for example cars, moving trees etc, but also edges that are present almost all the time during high visibility. The idea here is to count how many of these edges that are always present are in the image to be classified. The method can be divided into the following parts:

- Determine which edges are the ones that are preserved throughout the data set of high visibility images. Each edge in the image will be given a weight proportional to the number of images out of the total where it is present in.
- Find edges in the picture to be classified.
- Compute a score based on a weighted sum of the edges that are found in the image to be classified.

To determine the weights, the daylight images from a day with high visibility (selected manually) were used. Then, a Canny edge detector (built-in method in Mathematica with standard options) is used, after which the edges are thickened to the 8 neighbouring points. Having done this for all the images, each pixel has either value 0 or 1 for each image. Then the average value through all images $e_{mean}(m, n)$ has to be computed for each pixel (m, n) , resulting in Figure 9.



Fig. 9: Mean values of the edges

The weight for the (m, n) 'th pixel is defined as

$$w(m, n) = e_{mean}(m, n)^2 / [\sum_{m,n} e_{mean}(m, n)^2]$$

In Figure 10 the weights for Kastrup Vest are displayed.

When an image ought to be classified, the edges can be computed using the built-in Mathematica function followed by the calculation of a visibility score s .

$$s = \sum_{m,n} E(m, n)w(m, n)$$

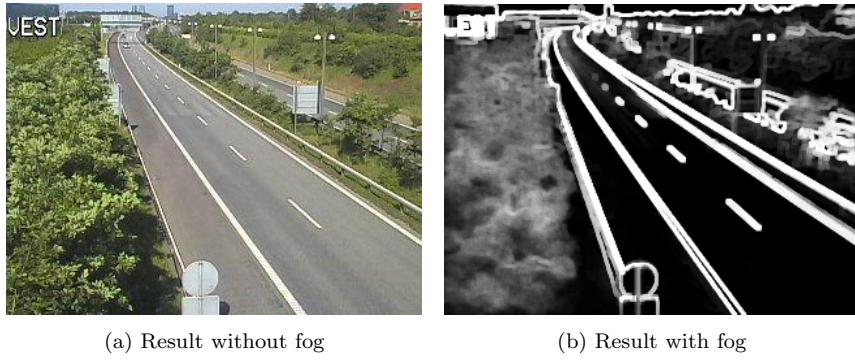


Fig. 10: Kastrup Vest image and resulting weights

where $E(m, n)$ is the edge value (0 or 1) for the (m, n) pixel. The score is then calculated for each image during that day and is shown in Figure 11. From the scores it can be inferred that the fog gradually disappears in the images from 1 to 30. Then, it stays clear for the rest of the day. However, a decrease in the score happens again after image 150.

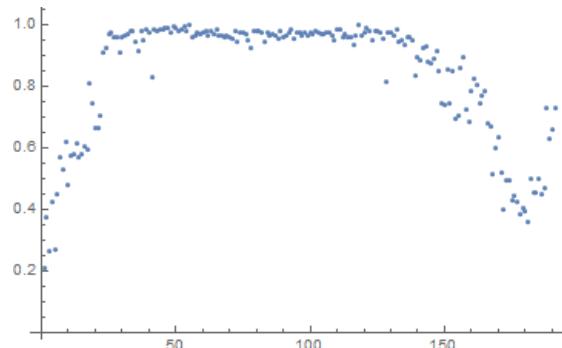


Fig. 11: Scores for June 17 between 4:05 and 19:55.

In order to check that the scores are consistent and to find out what is actually happening after image 150, a qualitative comparison between a few of the images for that day can be made by looking at Figure 12. The reason for the decrease in the score can be seen in image 180. It is due to strong artifacts from the sunshine. These images should be handled by other means.

In Figure 13, the scores are displayed for another day, June the 3rd, where there is only a little fog in the morning. And some of the images for this day are displayed in Figure 14. Again, it can be seen that the score fits the visibility conditions quite accurately.

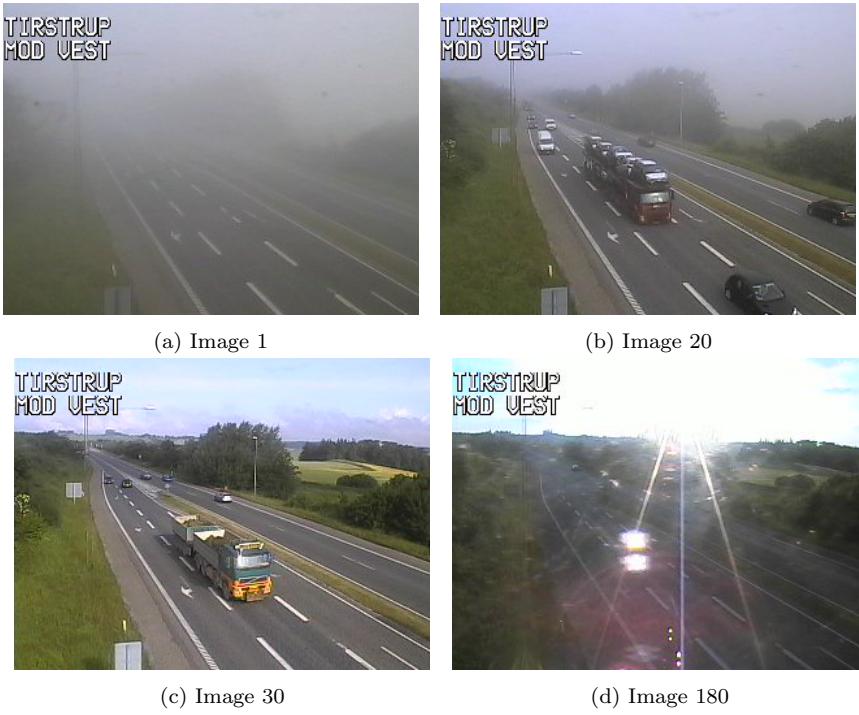


Fig. 12: Images 1, 20, 30 and 180 from June 17 corresponding to the score plot

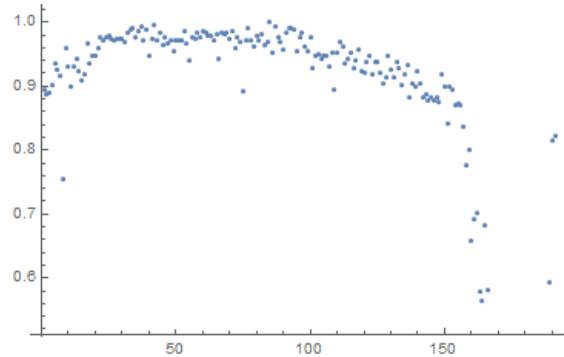


Fig. 13: Scores for June 3 between 4:05 and 19:55.

3.2 Classification

Each image needs to be classified as containing fog or not. This binary classification could be expanded later however, as for now, the labelled data only enables binary classification. As mentioned previously, annotation performed by experts/external measures were not available so some errors in the image labelling are expected.

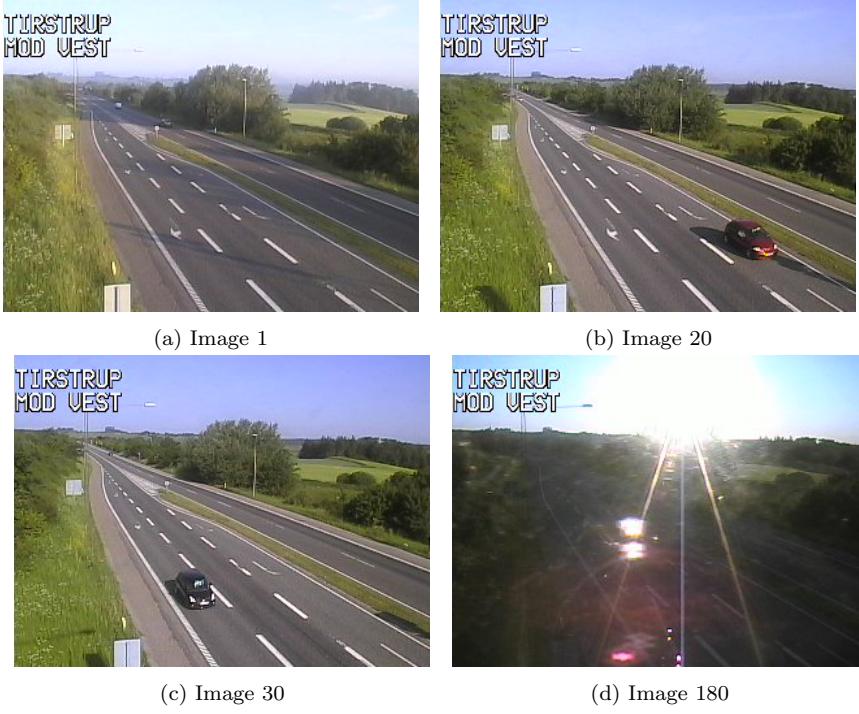


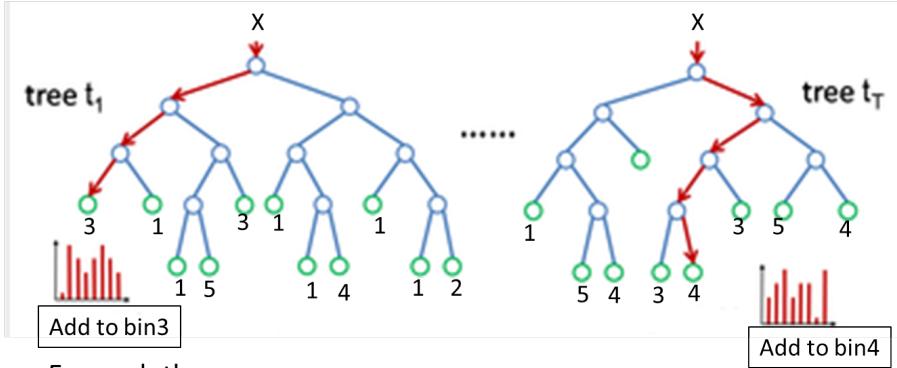
Fig. 14: Images 1, 20, 30 and 180 from June 3 corresponding to the score plot

The objective of a good classifier is to map the vector X^j to scalar Y_j , so that $Y_j = \bar{Y}_j$ where \bar{Y}_j is a reference categorical label of X^j .

A Random Forest (RF) classifier [1] is chosen to predict unseen images as containing fog $Y = 1$ or not $Y = 0$. Due to time constraints, RF was the only classification model investigated since it is the state of the art classifier that needs the least parameter tuning. To train the classifier the dataset ($n = 4781$) is split in two portions: a training set ($n = 3347$) and a test set ($n = 1434$). The image features extracted from the training set are used to train the RF classifier while the test image features are used to evaluate the accuracy of the classification model on images previously unseen by the model. The assignment of these images into training/test sets was done at random.

Roughly speaking, the RF works by combining a lot of weak classifiers into one strong classification. Each of these weak classifiers are called trees and, since a large number of trees make up a forest, the method gets the name random forest. The example forest in Figure 15 illustrates how a majority voting of all trees result in the forest decision. The number of trees that were used was 500, as it was found that more trees did not improve the results.

Four different models for the post-processing step of the features (see Figure 4) will be examined, as it seems from our experimental analysis that the features



- For each tree
 1. Push X down the tree
 2. Add three classification to histogram bin
- When done: Pick majority bin class

Fig. 15: Explains overall how a random forest works. Each feature vector X is passed down several trees. When X reaches a leaf of a tree, the leaf's label is added to a forest histogram. The forest decision is the label that grants the majority peak of the histogram.

need to be normalised so that the camera dependency is eliminated. The features where changing scale from camera to camera, see the jumps in Figure 16a for the sum of the Laplacian measure. After normalising, Figure is obtained instead 16b. Moreover, it is relevant to test whether it is useful to remove the parts of the image that are far way from the horizon and too close to the camera. This is due to the fog present in the very far distance not being relevant for the visibility and the latter is due to the fact that it has to be extremely foggy to alter the visibility for the objects that are very close. This removal was done by cropping the images.

Therefore the investigated models are:

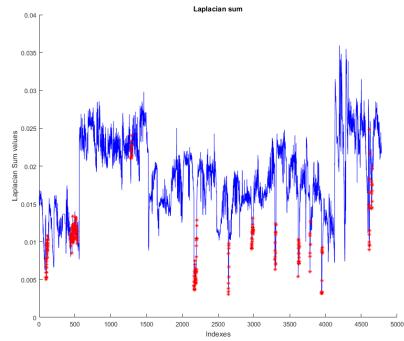
1. RF with full images and no normalisation
2. RF with full images and normalisation
3. RF with horizon cropped and no normalisation
4. RF with horizon cropped and normalisation

4 Results

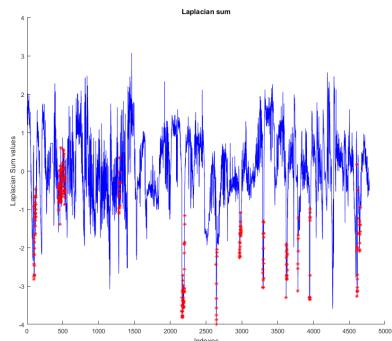
The success of these experiments is measured based on the classification performance over the test set. Then, the four different models can be compared.

4.0.1 Full image and no normalisation

This is the simplest way of handling the data. A classification is then trained, such that the predictive strength can be evaluated. From the test set an accuracy measure and confusion matrix is computed. In Table 1 The confusion matrix can



(a) Image 1



(b) Image 20

Fig. 16: The Laplacian absolute sum feature. Relative low values means fog and High means high visibility. Red markings indicate that fog is present. Note the scaling across different cameras. Camera1x: (image 1 to image 561), Camera1xx: (image 562 to image 1524), Camera2: (image 1525 to image 3944), Camera3: (image 3945 to image 4127), Camera4: (image 4128 to image 4781)

be seen and the accuracy was 0.9826. A large amount of times the prediction was no fog where there was actual fog. This is often caused by confusing weather conditions (sun streaks or borderline fog conditions).

Table 1: Confusion matrix for no normalisation and no cropping

	real Fog	no real Fog
predicted Fog	55	6
no predicted Fog	19	1353

A variable that quantifies the importance of each feature is produced when running the RF, results are in Figure 17. The dark channel prior carries most information for the RF classification model so removing this feature will hurt per-

formance most. However, most seem to contribute positively to the classification suggesting that features contain some complementary information.

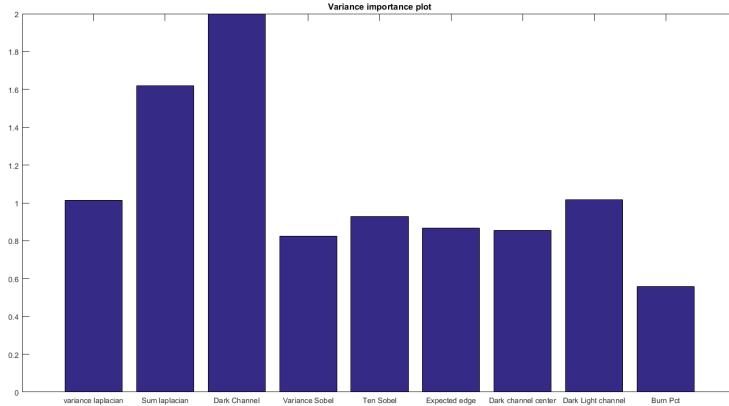


Fig. 17: Feature importance for the Random forest

4.0.2 Full images with normalisation

As previously mentioned, imposing normalisation is done to make features scale across the different cameras.

The classification accuracy was also 0.9826 here, with the confusion matrix seen in Table 2. The performance is like when not using normalisation, however on this model is likely to generalise for a new camera and new location, whereas without normalising it would need training data from the new camera.

Table 2: Confusion matrix for RF with normalisation

	real Fog	no real Fog
predicted Fog	55	6
no predicted Fog	19	1353

4.0.3 Cropped image with no normalisation

The cropped out images generally look as seen in Figure 18.

The accuracy in this case is 0.9830 and the confusion matrix is displayed in Table 3. These values are not considerably different from the previous two cases.

The feature importance is shown in Figure 19, the percentage of sunlight has less meaning than in the non-cropped cases. This shows that the feature is working as expected. Since this feature was created to deal with sun streaks, it is of lower



Fig. 18: Example image with the horizon cropped out.

Table 3: Confusion matrix for RF with no normalisation and on a cropped image

	real Fog	no real Fog
predicted Fog	56	6
no predicted Fog	18	1353

importance when the top part of the image is cropped, as most of the streaks disappear.

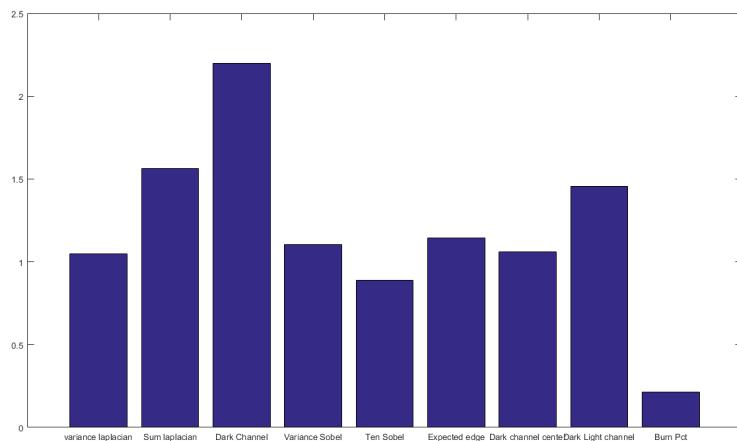


Fig. 19: Feature importance for the Random forest on cropped images

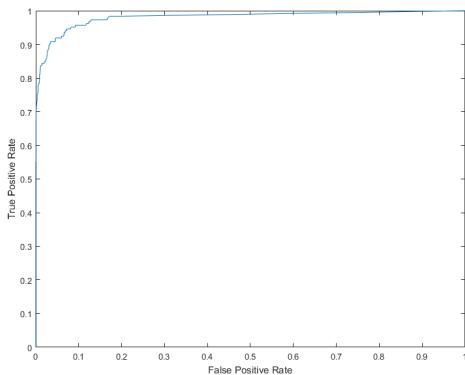


Fig. 20: ROC for the cropped and normalised images

4.0.4 Cropped image with normalisation

Here the accuracy is 0.9805 and the confusion matrix is in Table 4. The classification again seems to perform well but slightly worse than without normalisation, so again it is a question of whether we want a camera specific model or a general one.

Table 4: Confusion matrix for RF with normalisation and on a cropped image

	real Fog	no real Fog
predicted Fog	56	6
no predicted Fog	18	1353

4.0.5 Summary of Experiments

It was found that feature normalisation had a negative impact on classification, decreasing at most by less than a percent (0.9%). However it makes the features camera independant, which would be of interest if the method should be applied to new cameras without extra training. Furthermore, cropping the images did not change the classification results very much either. Overall all the models performed very similar. For comparison, an ROC curve was constructed for each experiment, but there were no visible differences here wither. In Figure 20 the ROC curve can be seen for one of the models. This curve can be utilised to change ones acceptance rate to either be more false positive sensitive or true positive sensitive by changing the threshold on the probability for each class in the random forest. In this case, one can see that to have a true positive rate of 95%, one will have to accept a false positive rate of 8%.

To better understand what leads to misclassifications, an investigation of the false positive and false negatives is performed on the non-cropped images. In Figure 21, four images which were classified as containing fog when labelled as not



Fig. 21: Four images classified as fog, when labelled as not fog

having fog are seen. The first and last two images probably contain fog in the distance, however it does not affect visibility in this case so they were labelled as not containing fog. The second image is clearly an artifact, as an effect of the sun streak blurring the image, which blurs the image and therefore grants smoother edges as if there was fog.

In Figure 22 four images that were classified as not foggy while labelled as foggy, that is false negatives. Here it is again clearly border examples, where non-experts might not be able to accurately label these images correctly. This again points at the need for more labelled data, such that the labels are properly set and the threshold between the decision fog no fog is better defined.

4.0.6 Alternatives to cropping

The goal of cropping is largely to remove non-stationary parts of the scenery from having a noisy influence on the fog detection. This was the reason for both cropping around the horizon and using the edge based feature where stationary edge image were given a higher weight for the decision. Another idea for getting rid of none stationary image parts could be applying a change detection algorithm. Due to time constraints, this approach was not fully implemented. However, this approach to obtain a mask that selects the stationary parts will be explained here. An example mask obtained from two images is shown in Figure 23.

The implementation of the change detection algorithm consist of 4 steps

- Global image normalisation:

Ensures that image 1 and image 2 both follow a normal distribution with mean 0 and standard deviation 1. Achieved by performing this pixelwise transformation to each image, $I_{norm} = (I - \mu(I)) / \Sigma_I$ where μ is the mean of the original



Fig. 22: Four images classified as not fog, when labelled as fog

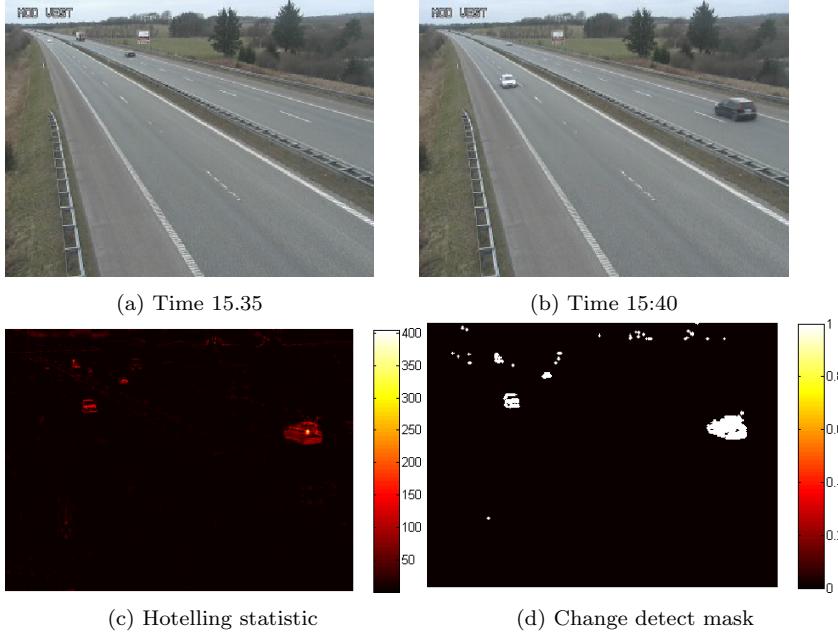


Fig. 23: Results from estimating a change detection mask using the between two neighbouring time points in the camera sequence. Here we show both two neighbouring images of the camera, a change detection statistics image and the final mask obtained from thresholding the statistic image using threshold $\zeta = 50$

image, Σ_I the covariance between the RGB channels making it a 3 by 3 matrix and $Inorm$ is the normalised image.

– Statistical difference:

The next step is to estimate the image difference between neighbouring images and to decide when this difference is significant. Here we used the Hotelling's T-squared which is an extension of the univariate T-test when you have more than one variable, in this case we have 3 (RGB). Hotelling's T-squared is now used to calculate test sizes that can be thresholded to define regions where pixels change significantly between neighbour images. The test size calculated per pixel is $T^2 = (x - \mu) \Sigma_{rgb} (x - \mu)$ where Σ_{rgb} is the covariance of the channel-wise image difference, x is intensity difference of the 3 channels of a pixel and μ is the zero hypothesis so that $\mu = [0; 0; 0]$ in our case.

– Thresholding:

We threshold the statistical images at $T^2 > 50$ to produce a mask of significant pixelwise change.

– Mask dilation:

The final step of the mask production is to grow the mask pixel boundary by 1 which can be done using morphological image dilation. This dilation is important if one wants to use the gradient or Laplacian based features since their estimates are based on the values in a local pixel neighbourhood.

5 Discussion and Conclusion

Generally the accuracy of the predictions was very good and around $\sim 98\%$. However, such a high accuracy is an effect of the class imbalance, since most of the data is non foggy. In spite of this imbalance a good detection rate for the fog images was achieved, with a 95 % true positive rate against an 8 % false positive rate. Note that there is a trade-off which means that the two rates can be improved at the cost of each other. Consequently, one should decide whether it is better to have false alarms or missed detections of fog.

There are several possibilities for improvement such as adding a mask or some weights defined by the change detection algorithm explained in Section 4.0.6. Another issue that should be dealt with is the sun streaks, an idea would be to introduce a feature that accounts for the sun height or the time of the day and season. Several other problem dependant issues, such as those mentioned in Section 1, should also be addressed so as to get a more robust system.

Actually, the time series of predictions can also be used for improving the accuracy of the results. Moreover, the information from nearby cameras could be included using some scheme such as majority voting or averaging of the results provided by each individual camera.

In the danish road camera system there are 500 cameras deployed over the more than 70 000 km of danish roads. This means that, if the cameras were uniformly distributed along the road network, there would be a separation of around 140 km between cameras. This is not the case, because the cameras are concentrated around Copenhagen, yet it is the countryside which is of special interest, as it is where there is less information due to the lack of people reporting or officers informing. As a consequence, the network of cameras is too sparse for cameras covering only a range of 25 km^2 . Thus, it is unlikely that there will be nearby

cameras from which a more accurate prediction of the visibility conditions can be performed. Nevertheless, there might be some specific areas in which cameras are close enough and a multi-camera system can be of use.

In the future, it would be interesting to look into the night issue. The method used here should be different and will detect fog using the lighting in the images (e.g. street lights, cars,...). In contrast, at some points in the night it is completely dark, so an idea would be to predict the fog using nearby cameras or the same camera but at other close by timings.

Furthermore, it should be noted that accurate labelling by experts or sensors such as scaterometres could provide a solid reference which is essential for an accurate classification.

References

1. Leo Breiman. Random forests. *Mach. Learn.*, 45(1):5–32, October 2001.
2. Bolun Cai, Xiangmin Xu, Kui Jia, Chunmei Qing, and Dacheng Tao. Dehazenet: An end-to-end system for single image haze removal. *arXiv preprint arXiv:1601.07661*, 2016.
3. Laurent Caraffa and Jean Philippe Tarel. Daytime fog detection and density estimation with entropy minimisation. *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, 2(3):25, 2014.
4. Robert G. Hallowell, Michael P. Matthews, and Paul A. Pisano. Automated extraction of weather variables from camera imagery.
5. Kaiming He, Jian Sun, and Xiaou Tang. Single image haze removal using dark channel prior. *IEEE Trans. Pattern Anal. Mach. Intell.*, 33(12):2341–2353, December 2011.
6. Wei Huang and Zhongliang Jing. Evaluation of focus measures in multi-focus image fusion. *Pattern Recogn. Lett.*, 28(4):493–500, March 2007.
7. Johannes Kopf, Boris Neubert, Billy Chen, Michael Cohen, Daniel Cohen-Or, Oliver Deussen, Matt Uyttendaele, and Dani Lischinski. Deep photo: Model-based photograph enhancement and viewing. In *ACM Transactions on Graphics (TOG)*, volume 27, page 116. ACM, 2008.
8. Sungmin Lee, Seok Min Yun, Ju-Hun Nam, Chee Sun Won, and Seung-Won Jung. A review on dark channel prior based image dehazing algorithms. *EURASIP J. Image and Video Processing*, 2016:4, 2016.
9. J. L. Pech-Pacheco, G. Cristobal, J. Chamorro-Martinez, and J. Fernandez-Valdivia. Diatom autofocusing in brightfield microscopy: a comparative study. In *Pattern Recognition, 2000. Proceedings. 15th International Conference on*, volume 3, pages 314–317 vol.3, 2000.
10. José Luis Pech-Pacheco, Gabriel Cristóbal, Jesús Chamorro-Martínez, and Joaquín Fernández-Valdivia. Diatom autofocusing in brightfield microscopy: a comparative study. In *Pattern Recognition, 2000. Proceedings. 15th International Conference on*, volume 3, pages 314–317. IEEE, 2000.
11. Said Pertuz, Domenec Puig, and Miguel Angel Garcia. Analysis of focus measure operators for shape-from-focus. *Pattern Recognition*, 46(5):1415–1432, 2013.

12. P. F. Alcantarilla S. Bronte, L. M. Bergasa. Fog detection system based on computer vision. *Proceedings of the 12th International IEEE Conference on Intelligent Transportation Systems, St. Louis, MO,,* 2009.