Weather Classification with traffic surveillance cameras

Khouloud Dahmane [1,2*], Pierre Duthon [1], Frédéric Bernardin [1], Michèle Colomb[1], Christophe Blanc [2], Frédéric Chausse [2]

[1] Cerema Centre-Est, Department Laboratory of Clermont-Ferrand, 8-10 rue Bernard Palissy, F-63017 Clermont-Ferrand Cedex 2, France
[2] Institut Pascal, UMR 6602 UCA/CNRS/Sigma, Campus des Cezeaux, 4 impasse Blaise Pascal, CS 60026. 63178 Aubiere Cedex, France
*dahmane.khouloud@gmail.com

Road operators are using Intelligent Transport Systems composed of road side cameras for traffic management. The artificial vision algorithms used for automatic detection may be impacted by adverse weather conditions. Therefore, it is necessary to improve these algorithms in such conditions. In addition, the applications developed to operate in road context impose a perfect reliability of operation including in adverse weather conditions. There are many works that allow weather classification but they do not take into consideration all the degraded conditions. In this paper, we propose a method based on convolutional neural networks to classify adverse weather conditions from a road camera. This method could be extended to on-board cameras used by autonomous vehicles. We also present the weather image databases that we use to evaluate our learning.

1. Introduction

ITS are usually used in favourable weather conditions where they show their performance and reliability. But it turned out that vision systems such as video surveillance and advanced driver-assistance systems (ADAS) are generally affected by the degradation of weather conditions (fog, heavy rain, etc.) due to limited visibility and reduced image quality [1], [2]. Autonomous driving is a national and european topic [3], [4] and it will requires research and inovation on the specific use cases, critical for safety as adverse weather conditions. The French project “Nouvelle France Industrielle Véhicule Autonome” dedicated to autonomous vehicles is focused on the study of road security and operating safety for autonomous vehicles and has shown that input data such as sensors, maps, and signs are insufficient to ensure road safety. Also, the methods and the tools of design and operating safety are insufficient where they meet limitation of performance and uncertainties of detection or recognition, localization and navigation. Finally, the existing simulation methods and tools do not take into account the particular issues of operating safety.

In order to optimize the vision systems performance, it is necessary to have a reliable detection system for the presence of adverse weather conditions.

There are dedicated meteorological sensors for physical measurement, but they are expensive. As cameras are already installed on the road, they can be used to satisfy two functions at once: image acquisition and physical measurement instead of dedicated weather sensors. This is the aim of our study "Weather classification with traffic surveillance cameras".

There are some works that distinguish between different weather conditions which are based on extracting image features or using image descriptors. In a road context, vision systems are generally based on cameras having a final function such as obstacle detection, in the case of on-board cameras, or automatic detection of incidents, in the case of road cameras. These cameras use image descriptors. The latter can be disturbed by adverse weather conditions. In this case, it will be interesting that the cameras can self-diagnose. In [5], [6] the authors explain this idea. Indeed, if there are descriptors sensitive to the rain or the fog and which are used in a final function, and if in addition the cameras know how to measure the quantity of the present meteorological perturbation, they signify that there is a disruption of the final function (eg pedestrian detection for on-board cameras). In [7], [8], [9] the authors measure rainfall intensity by artificial vision systems using segmentation. Indeed, by doing a background extraction and comparing successive images, they manage to detect moving objects. If these movements are regular and in the same way, it must be rain. In this case, they can measure the rain.

Other works classify the weather but with a limited number of classes. The authors of [10] categorize two weather classes sunny and cloudy using weather dataset containing 10,000 single images. They are based on the extraction of weather cues for classification, such as sky, shadow, reflection, contrast and haze. They used collaborative learning to successfully classify weather which is based on homogeneous voters. On the other hand, in [1] the authors classify weather according to 3 classes in order to recognize the different intensities of rain: clear, light rain and heavy rain. They use a database containing 500,000 single monoscopic color images from in-vehicle camera. They propose a method that gathers several histogram features in a vector such as brightness, contrast, sharpness, saturation and hue and make classification with linear SVM. Also, the authors of [2] used 3 classes of weather: sunny, cloudy and rainy to classify 2496 images acquired from video captured by vision system in vehicle. They are based on local and global descriptors such as HSV, HGA and Road. They use Real AdaBoost as classifier.

Although previous works have proposed interesting solutions in weather classification, they do not take into consideration all possible weather conditions including fog or night conditions.

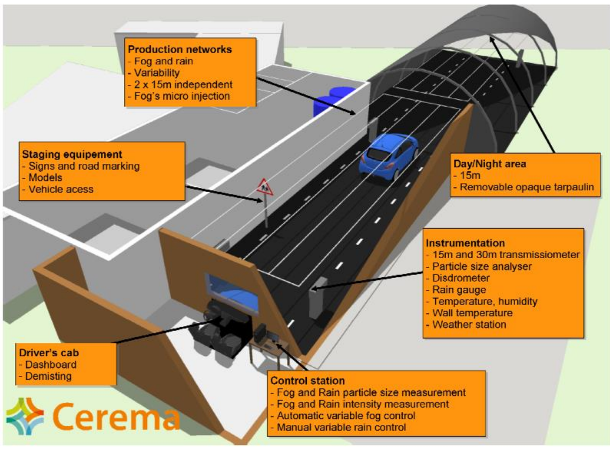


Fig. 1 *The Cerema's Fog&Rain R&D platform [22]*

Interested by the success of convolutional neural networks (CNN) in the field of vision and more precisely in image classification and recognition [11], [12], [13], [14], [15] we adapted the deep learning method to solve the problem of weather classification. In [16] the authors used CNN and did learning on the same database as [10]. The authors of [17] predict the ambient temperature, the season, the month, the week and the day from a given image. The learning is done on AMOS database [18] which contains over one million images acquired from different webcams located everywhere in the world. [19] classifies weather according to 4 classes sunny, rainy, snowy and haze using MWI database which contains 20 000 images of different outdoor scenes. The authors used multiple category-specific dictionary learning and multiple kernel learning and extracted local features such as sky, shadow, rain streak, snowflake and dark channel and global descriptors such as contrast and saturation.

In this paper, we will deal with, in addition to normal conditions, the case of rain and fog. Our study takes into consideration the day and the night. In order to evaluate our learning and labelling strategy, we create a large database of images acquired by a road camera and dedicated sensors for meteorological measurements.

The key contribution of this paper is the learning transfer method [20]. Indeed, this method could be useful for ITS applications. It involves using input data from a certain space to train a CNN and allows the weather classification in another space containing different input data to the first one. In this case, if learning transfer is done successfully, we can classify weather on sites that do not contain weather sensors by using only road cameras.

The next section presents a description of two weather databases that will be used for our experiments. In section 3, the used convolutional neural network is presented. Section 4 showed detailed experimental classification results obtained on our weather databases. Finally, we conclude the paper in section 5.

2. Datasets

2.1. Technical settings

Our two databases are acquired on two different areas.

The first one called Cerema's Fog&Rain Research & Development platform [21], [22] (see Fig. 1). It is the only place in Europe producing controlled adverse weather conditions such as rain and fog. This is a 31-meter long platform divided into two parts: a tunnel (15 m length and 5.5 m width) of durable construction and a greenhouse (16 m length and 8.5 m width) of lightweight construction, maintained by two arches and covered by two sheets, one black and one transparent (to provide night-time and daytime conditions respectively). This platform is equipped with several sensors. The meteorological visibility distance is measured with the Degreane Horizon TR30 transmissometer (see Fig. 2.c). This sensor allows the measurement of the meteorological visibility over the measurement range useful for the road context (between 5 m and 1 000 m). Rain intensities are measured using the LSI DQA136 tipping bucket rain gauge and the OTT Parsivel optical disdrometer (see Fig. 2.a and Fig. 2.b) which measures rainfall intensities between 0.001 mm/h and 1200 mm/h. The database was built using Sony DFW-X700 camera with a resolution of 1024 x 631 at a frequency of 7.5Hz. The camera was positioned 120 centimetres above the ground, which corresponds to a driving situation.

(a) (b) (c) (d)

Fig. 2 *Weather condition measurement sensors. (a) OTT Parsivel optical disdrometer, (b) LSI DQA136 tipping bucket rain gauge, (c) Degreane Horizon TR30 transmissometer, (d) Vaisala PWD12 weather sensor.*

The second area is called Fageole highway on the E70 motorway which belongs to “Direction interdépartementale des routes Massif-Central” (see Fig. 3). It is an outdoor site that contains weather sensors such as the Precis Mecanique 3039 tipping bucket rain gauge which measures rainfall intensities between 0.1 mm/h and 1000 mm/h and the Vaisala PWD12 weather sensor (see Fig. 2.d) which makes it possible to measure the visibility between 10 m and 2 000 m. The database was created by AVT Pike F421B camera with a resolution of 1008 x 648. It is positioned 4.5 m above the ground.

Fig. 3 *Fageole highway weather station on the E70 motorway*

Fig. 4 *A selection of images from the Cerema-AWP database under different conditions. Day conditions and night conditions. From left to right: Normal conditions, Fog 1, Fog 2, Rain 1, Rain 2 [23].*

2.2. Cerema-AWP Database

The Cerema Pedestrian database [23] which will be called Cerema-AWP database (Cerema Adverse Weather Pedestrian database) has taken place at the first time for pedestrian detection. But since it contains different weather conditions, we take it into account for our study. It is a weather image database consisting of 62,828 images acquired at Cerema's Fog&Rain R&D platform [21], [22]. This database is divided into 10 sets acquired by day and by night. For these both conditions, there are 2 rain intensities and 2 fog densities.

For day light fog (DF1) and day heavy fog (DF2) the visibilities are respectively between 70 m and 80 m and between 50 m and 60 m. For day light rain (DR1) and day strong rain (DR2) rainfall intensities are respectively between 20 mm/h and 30 mm/h and between 40 mm/h and 50mm/h.

The distributions of rain and fog intensities are described in Fig. 5.

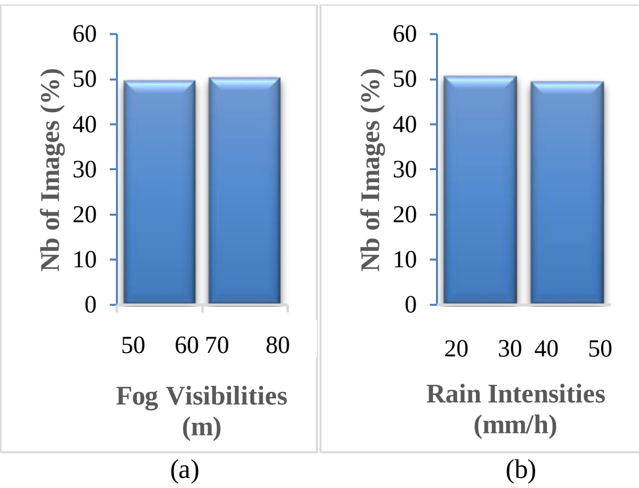
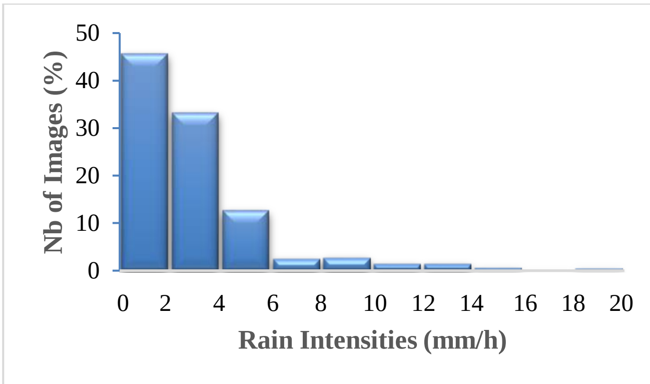


Fig. 5 *The distribution of fog visibilities (a) and rain intensities (b) on the Cerema-AWP Database*

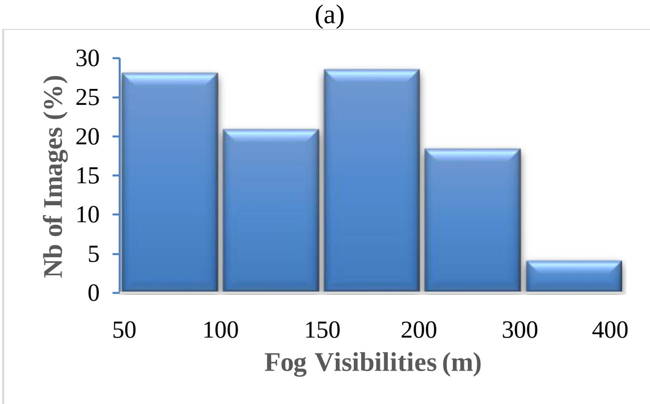
2.3. Cerema-AWH Database

The Cerema-AWH database (Cerema Adverse Weather Highway database) contains today more than one million images, acquired on the Fageole highway during day and night. The images are in grey scale. In our study, we will use about 340K images acquired between February 2, 2017 and May 6, 2017. Labelling is done automatically by associating each image with the corresponding weather data. These data are recorded from the meteorological sensors located on the Fageole highway.

The weather conditions presented in the database are: rain, fog and snow. Some example images from each condition are shown in Fig. 7. These conditions are real conditions (unlike those of Cerema-AWP Database [23] which are simulated). The rainfall intensities used in our experimentations vary between 2 mm/h and 90 mm/h but Fig.7.a presents only intensities between 0 mm/h and 20 mm/h. The fog visibilities studied vary between 50 m and 400 m (see Fig. 7.b).



(a)



(b)

Fig. 6 *The distribution of rain intensities (a) and fog visibilities (b) on the Cerema-AWH Database*

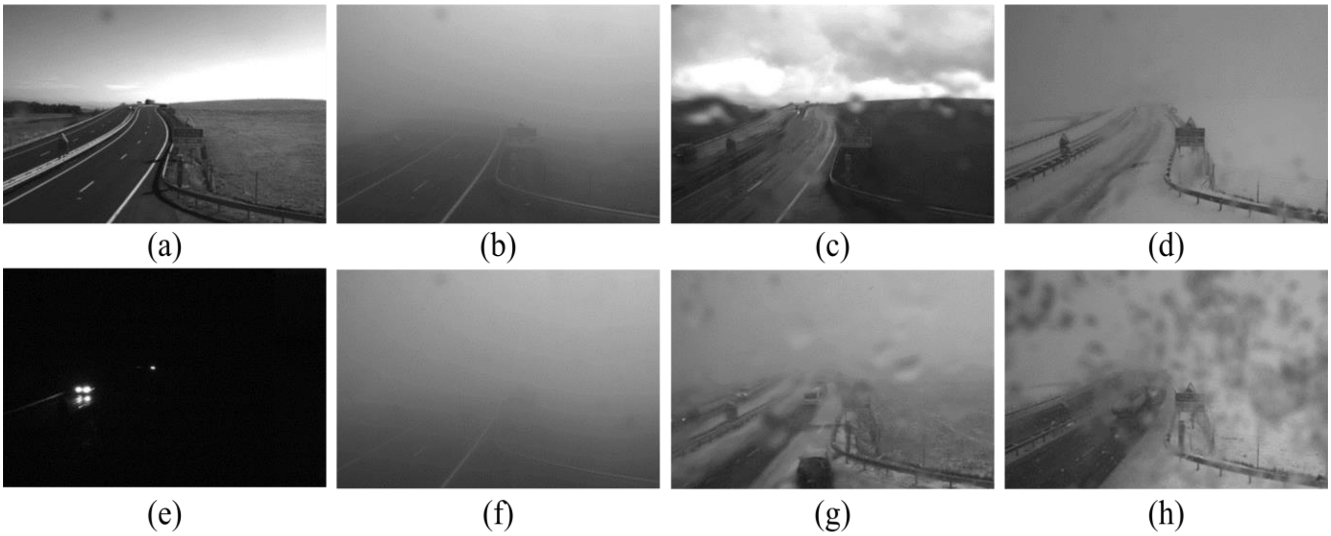


Fig. 7 *A selection of images from the Cerema-AWH database under different conditions. Day conditions (a) and night (e) conditions. Light fog (b) and heavy fog (f). Light rain (c) and heavy rain (g). Light snow (d) and heavy snow (h).*

3. Convolutional neural networks

We use the LeNet-5 [24] for our convolutional neural network. It was one of the first convolutional neural networks that helped propel the field of deep learning. It has been designed by Yann LeCun since 1998 [24]. At that time, LeNet-5 architecture was used primarily to recognize numbers and applied to the recognition of handwritten numbers on checks.

In general, LeNet-5 architecture contains a convolution layer followed by a pooling layer, another convolution layer followed by a pooling layer, and then two fully connected layers similar to conventional multilayer perceptrons (see Fig. 8).

We will use a slightly different version of the original LeNet-5 implementation, replacing sigmoid activations with rectified linear unit (ReLU) activations for neurons to enhance training. Indeed, in all experiments, researchers found that networks based on rectified linear units performed better than networks with sigmoid-type activation functions [25]. Most deep learning algorithms try to achieve model optimization. This optimization is done by minimizing the error or the loss function using the gradient method.

The gradient method makes it possible to modify each time the weights of the neurons. Indeed, the neural network is composed of a succession of layers, each one takes its inputs on the outputs of the previous one. Each layer (j) is composed of N_j neurons, taking their inputs on the N_{j-1} neurons of the previous layer. Each synapse is associated with a synaptic weight so that the N_{j-1} are multiplied by this weight, then added by the level j neurons, which is equivalent to multiplying the input vector by a transformation matrix.

The output associated with the inputs x_1 to x_n is written in the equation (1).

$$o = \varphi\left(\sum_{i=1}^{n} w_i x_i + b\right) \qquad (1)$$

where n is the number of inputs, φ is the activation function and b is the bias.

Neural networks often use a gradient descent on weights. This means at each iteration, the backpropagation calculates the derivative of the error (also called the loss function) with respect to each weight and subtract it from that weight. Indeed, we try to minimize $L(W)$ where W is the parameter vector which includes weights and the classical gradient method is written in (2).

$$W_{t+1} = W_t - \rho L(W_t) \qquad (2)$$

where $\rho > 0$ is the step of the iterative gradient method.

However, following this equation, the weights will change too much at each iteration, allowing the error to increase or diverge. In practice, researchers typically multiply each derivative by a small value called the learning rate before subtracting it from its corresponding weight. This sets up the stochastic gradient descent (SGD).

In other words, the stochastic gradient descent updates the weights W by a linear combination of the negative gradient ∇L(W) and the update of the previous weight V_t. The learning rate α is the negative gradient weight. The momentum μ is the weight of the previous update.

Formally, we have the following equations (3) and (4) to calculate the updated value V_{t+1} and the updated weights W_{t+1} at the t + 1 iteration, given the previous weight update V_t and current weights W_t.

$$V_{t+1} = \mu V_t - \alpha \nabla \mathrm{L}(W_t) \qquad (3)$$
$$W_{t+1} = W_t - V_{t+1} \qquad (4)$$

In the solver parameters, the learning rate is variable and decreases with each iteration according to the following equation (5).

$$\alpha = \alpha_0\,(1 + \gamma n)^{-p} \qquad (5)$$

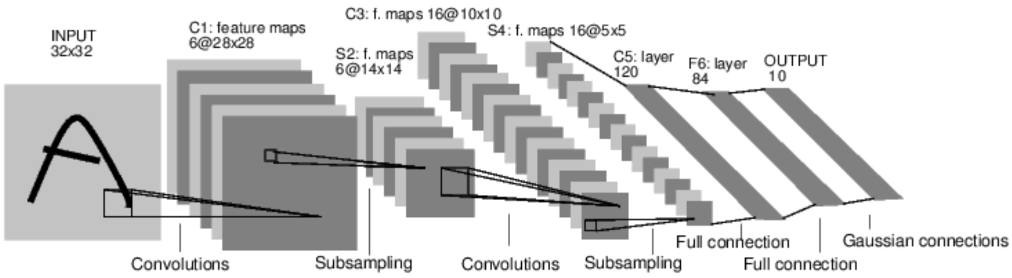


Fig. 8 *LeNet-5 Convolutional Neural Network Architecture [24]*

where n is the current iteration and the hyper-parameters: learning rate α_0, gamma γ and the power p are to be defined in the solver.

4. Results and experiments

In this section, we will present our method and report the classification results obtained. We will use the two weather databases as described in the section 2.

We use Caffe Deep Neural Network tool [26] for training and testing all our experiments. We run them in Python code on a GPU of GeForce GTX 1080 with 8G RAM.

4.1. Classes number influence

In this part, we will study the influence of the number of classes on the classification accuracy.

For the first test, we have done the classification according to 4 weather classes: day fog (DF), day rain (DR), day normal conditions (DNC) and night normal conditions (NNC). Normal conditions are included to provide a reference state for measuring only the impact of degraded conditions.

For DF we chose images with visibility less than 400m. For DR rainfall intensity is greater than 2 mm/h. For DNC and NNC, the visibility is higher than 1000 m.

For the day conditions, the images are acquired between 8 am and 5 pm, and for the conditions of night, the images are acquired between 10 pm and 6 am.

The learning database contains 12 000 images and the testing database contains 1200 images. We have an equal amount of images for each class.

In order to have reliable classification results, we ensured that no image is used for both training and testing databases. For this, we have used, for the learning one, images acquired during the even days and for the testing one, images acquired during odd days.

The learning is done with 20 000 iterations with a learning rate α_0 set to 0.01 and a gamma γ set to 0.001. The momentum μ is set to 0.9 and weight decay to 0.0005.

For input data, we used 28x28 contrasted patch. It is a part of the roadway containing the black road and a white marking (Fig. 9).

We execute 10 rounds and report the higher classification accuracy which is equal to **99.7 %.**

As our initial aim is measuring weather by camera for classification, we thought to refine the classification and increase the number of classes. In this case, for the 2nd test, the classes correspond to weather situations are C = {DF1, DF2, DR1, DR2, DNC, NNC} which contain 2 fog densities (DF1 and DF2) and 2 rain intensities (DR1 and DR2).

Based on the distribution of fog visibilities (Fig. 6.b) and rainfall intensity (Fig. 6.a) in the Cerema-AWH database, we have chosen the characteristics of the 4 weather classes. For light fog (DF1), the visibilities vary between 150m and 400m. For heavy fog (DF2), visibilities vary between 50m and 150m. Light rain (DR1) contains intensities between 2 mm/h and 4 mm/h and heavy rain (DR2) intensities are greater than 4 mm/h.

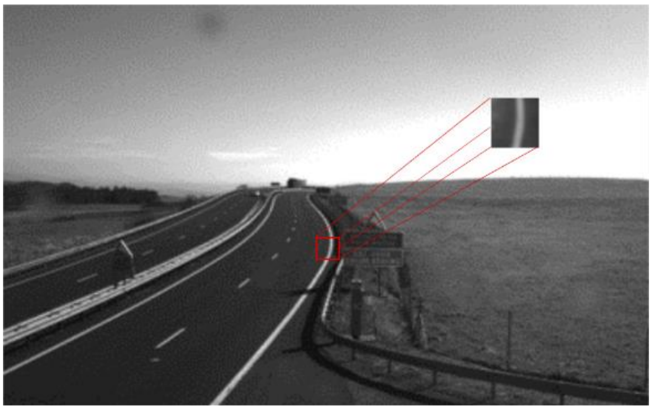


Fig. 9 *The contrasted patch (size 28 x 28)*

In total there are 24 000 images, learning base contains 16 000 images and testing database contains 8 000 images. To be able to compare the two tests, we used the same learning parameters.

Despite the difficulty of the task, CNNs perform well in weather classification, it achieves **77.0 %** of classification accuracy for a more detailed classes.

We note that for a rough classification (4 weather classes) we have a very good classification accuracy. On the other hand, if we think to make a more refined classification (6 weather classes) we will have less good accuracies. In this case, we must improve these last results. It will be the aim of future work by using other networks, or greater number of layers.

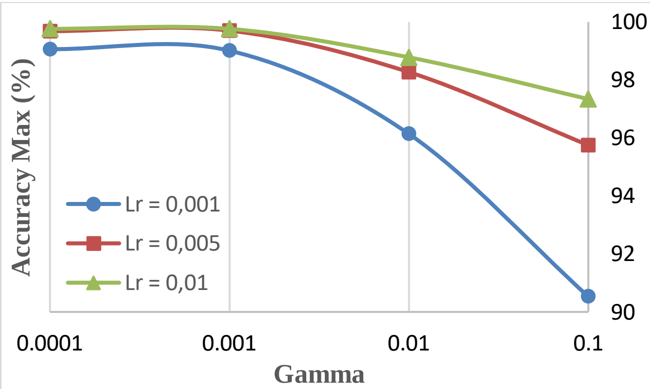
4.2. Hyper-parameters influence

In this part, we will study the influence of changing of learning parameters on the classification results. To do this, we will apply each time a couple of learning rate and gamma. We test 3 different α_0 values in equation (5), i.e., {0.001, 0.005, 0.01} and 4 different γ values in the same equation, i.e., {0.0001, 0.001, 0.01, 0.1}.

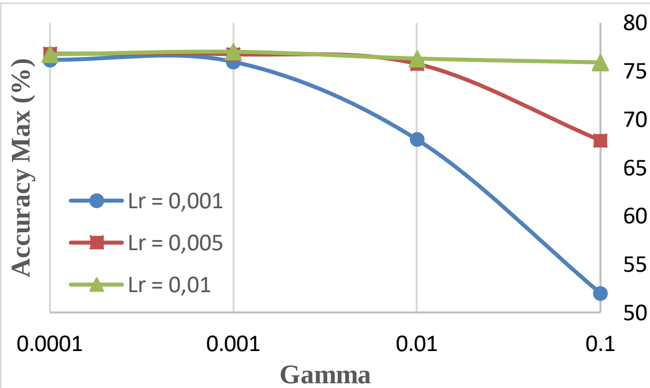
We started learning with different hyper-parameters couples on the Cerema-AWH database with 4 and 6 weather classes. Fig. 10 shows classification accuracy for each value of learning rate and gamma. For each couple of hyper-parameters, we execute 10 rounds and report the higher classification accuracy.

We notice that the more we increase the learning rate the more we increase the accuracy. On the other hand, the more one decreases the gamma the more we increase the accuracy.

For the same couple of parameters, we have the same behaviour for both tests. In this case, we can fix the couple with which we will follow our learning. Also, we must avoid working with low learning rate and high gamma.



(a)



(b)

Fig. 10 *Results of learning with different hyper-parameters couples on Cerema-AWH database. (a) 4 weather classes and (b) 6 weather classes*

4.3. Learning transfer

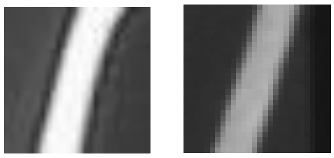
In the previous parts, we have done the weather classification on the Cerema-AWH database and we obtained good results.

For that, we thought to use another database with other characteristics. Learning and testing on the same database is a good idea and has worked well. But what about using two databases at the same experiment?

In this part, we will learn on the Cerema-AWP database (Section 2.1) and test on the Cerema-AWH database. It is the learning transfer.

We will apply the CNN to classify each image into one of the classes C = { DF, DR, DNC, NNC}, during 20 000 iterations, with a learning rate α_0 set to 0.001 and a gamma γ set to 0.001. The momentum μ is set to 0.9 and weight decay to 0.0005.

For the input data, we have chosen similar patches of size 28x28 (see Fig. 11) so that the neural network only takes into account the change in adverse weather conditions. The images of the training database are randomly shuffled before feeding the CNN.



(a) (b)

Fig. 11 *Similar Patch images. Train patch from Cerema-AWP Database (a) and Test patch from Cerema-AWH Database (b)*

We learn and we get a classification accuracy of **65.7 %**. It is noted that the accuracy is worse compared to the one obtained on the Cerema-AWH database for 4 weather classes (**99.7 %**).

We investigate the classification results in Table 1 in more detail to find out which weather conditions are miss-classified.

	DF	DNC	NNC	DR
DF	2 506	18	0	1 492
DNC	0	1 406	67	4
NNC	0	1	3 616	0
DR	1 194	2 275	12	2 204

Table 1 *Classification results at 20K iterations on 14 795 images with accuracy of 65.7 %*

Table 1 shows the confusion matrix of the 4 weather classes of the learn transfer from Cerema-AWP database to Cerema-AWH database. We can see that 1194 images of DF, 2275 images of DNC and 12 images of NNC are predicted as DR. Also there are 1492 images of DR are recognized as DF. So it is clear that the CNN has a problem with the classification of day rain (DR).

For that, we check rain images misclassified. It turned out that for the Cerema-AWH database, the raindrops stick on the lens of the road camera, unlike the rain images acquired at the Cerema's Fog&Rain R&D platform.

Following this problem, we omit rain images from the two databases to better compare them and we run tests again. In this case, and with 3 weather classes C = {DF, DNC, NNC}, we obtained a classification accuracy of **96.8 %**.

Despite all the following differences:

- the Cerema-AWP database contains images acquired in an indoor site (Cerema's Fog&Rain R&D platform) and the Cerema-AWH database contains images acquired in an outdoor site,
- the Cerema-AWP database contains simulated weather conditions and the Cerema-AWH database contains real conditions,
- the fog visibilities in the Cerema-AWP database vary between 50m and 80m and the fog visibilities

in the Cerema-AWH database vary between 50 m and 400 m,

we managed to have a good classification accuracy of **96.8 %** by the learning transfer method.

5. Conclusion

In this work, we present a two weather databases Cerema-AWP database and Cerema-AWH database containing respectively indoor and outdoor images acquired under adverse weather conditions with an automatic labelling of images. We present an approach that is able to distinguish between 4 and 6 types of weather based on Convolutional neural networks and our experimental results achieves a high classification accuracy of more than 99.0 % for the distinction between 4 weather classes. CNNs proved that they are successful in turning a camera into weather sensors. Similarly, we present results of the learning transfer method between two different databases. The results that we achieved proved that this method can be effective and used in many ITS applications.

We expect better performance in the future by adding other weather conditions like snow. Improvements of the overall classification results could be achieved by applying other CNNs architectures like VGG or ResNet. We can also use recurrent neural networks that take into account the weather for classifying video sequences which can be interesting for video surveillance.

6. Acknowledgments

This work has been sponsored by the French government research program "Investissements d'Avenir" through the IMobS3 Laboratory of Excellence (ANR-10-LABX-16-01) and the RobotEx Equipment of Excellence (ANR-10-EQPX-44), by the European Union through the Regional Competitiveness and Employment program 2014-2020 (ERDF - AURA region) and by the AURA region.

7. References

[1] M. Roser and F. Moosmann, “Classification of weather situations on single color images,” 2008 IEEE Intelligent Vehicles Symposium, pp.798–803, Jun. 2008.

[2] X. Yan, Y. Luo, and X. Zheng, “Weather Recognition Based on Images Captured by Vision System in Vehicle,” Advances in Neural Networks – ISNN 2009, pp.390–398, May 2009.

[3] “Automated Driving Roadmap - Report from ERTRAC Working group, Connectivity and Automated Driving,” ERTRAC Edition, 29-May-2017.

[4] “Objectifs de recherche - Nouvelle France Industrielle ‘Vehicule Autonome,’” Objectifs-de-recherche-Vehicule-Autonome-4.pdf, Centre Pilotage, 16-Jul-2015.

[5] P. Duthon, F. Bernardin, F. Chausse, and M. Colomb, “Methodology Used to Evaluate Computer Vision Algorithms in Adverse Weather Conditions,” Proceedings, Transportation Research Arena TRA2016, Warsovie, Pologne, vol.14, avril 2016.

[6] P. Duthon, F. Chausse, F. Bernardin, and M. Colomb, “Descripteurs d’images en contexte routier : une revue bibliographique,” Conference: Reconaissance de Formes Intelligence Artificielle (RFIA), At Clermont-Ferrand (France), Jun. 2016.

[7] K. Garg and S. K. Nayar, “Detection and removal of rain from videos,” Proceedings of the 2004 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2004. CVPR 2004., vol.1, p.I-528-I-535 Vol.1, Jun. 2004.

[8] K. Garg and S. K. Nayar, “When does a camera see rain?,” Tenth IEEE International Conference on Computer Vision (ICCV’05) Volume 1, vol.2, p.1067–1074 Vol. 2, Oct. 2005.

[9] K. Garg and S. K. Nayar, “Vision and rain,” *Int. J. Comput. Vis.*, vol.75, no. 1, p.3, 2007.

[10] C. Lu, D. Lin, J. Jia, and C.-K. Tang, “Two-Class Weather Classification,” Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp.3718–3725, 2014.

[11] K. Simonyan and A. Zisserman, “Very deep convolutional networks for large-scale image recognition,” *ArXiv Prepr. ArXiv14091556*, 2014.

[12] R. Girshick, J. Donahue, T. Darrell, and J. Malik, “Rich feature hierarchies for accurate object detection and semantic segmentation,” Proceedings of the IEEE conference on computer vision and pattern recognition, pp.580–587, 2014.

[13] A. Krizhevsky, I. Sutskever, and G. E. Hinton, “ImageNet Classification with Deep Convolutional Neural Networks,” in Advances in Neural Information Processing Systems 25, F. Pereira, C. J. C. Burges, L. Bottou, and K. Q. Weinberger, eds. Curran Associates, Inc., 2012, pp.1097–1105.

[14] P. Sermanet, D. Eigen, X. Zhang, M. Mathieu, R. Fergus, and Y. LeCun, “Overfeat: Integrated recognition, localization and detection using convolutional networks,” *ArXiv Prepr. ArXiv13126229*, 2013.

[15] J. Donahue, Y. Jia, O. Vinyals, J. Hoffman, N. Zhang, E. Tzeng, and T. Darrell, “Decaf: A deep convolutional activation feature for generic visual recognition,” International conference on machine learning, pp.647–655, 2014.

[16] M. Elhoseiny, S. Huang, and A. Elgammal, “Weather classification with deep convolutional neural networks,” 2015 IEEE International Conference on Image Processing (ICIP), pp.3349–3353, Sep. 2015.

[17] A. Volokitin, R. Timofte, and L. V. Gool, “Deep Features or Not: Temperature and Time Prediction in Outdoor Scenes,” 2016 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), pp.1136–1144, Jun. 2016.

[18] N. Jacobs, N. Roman, and R. Pless, “Consistent temporal variations in many outdoor scenes,” Computer Vision and Pattern Recognition, 2007. CVPR’07. IEEE Conference on, pp.1–6, 2007.

[19] Z. Zhang, H. Ma, H. Fu, and C. Zhang, “Scene-free multi-class weather classification on single images,” *Neurocomputing*, vol.207, pp.365–373, Sep. 2016.

[20] S. J. Pan and Q. Yang, “A survey on transfer learning,” *IEEE Trans. Knowl. Data Eng.*, vol.22, no. 10, pp.1345–1359, 2010.

[21] M. Colomb, K. Hirech, P. André, J. J. Boreux, P. Lacôte, and J. Dufour, “An innovative artificial fog production device improved in the European project ‘FOG,’”

Atmospheric Res., vol.87, no. 3–4, pp.242–251, Mar. 2008.
[22] "Bilan du congrès mondial ITS2015 à Bordeaux," http://www.sud-ouest.cerema.fr/bilan-du-congres-mondial-its2015-a-bordeaux-un-a995.html, accessed Jan. 18. 2018. .
[23] K. Dahmane, N. E. B. Amara, P. Duthon, F. Bernardin, M. Colomb, and F. Chausse, "The Cerema pedestrian database: A specific database in adverse weather conditions to evaluate computer vision pedestrian detectors," 2016 7th International Conference on Sciences of Electronics, Technologies of Information and Telecommunications (SETIT), pp.472–477, Dec. 2016.
[24] Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proc. IEEE*, vol.86, no. 11, pp.2278–2324, Nov. 1998.
[25] M. A. Nielsen, "Neural Networks and Deep Learning," http://neuralnetworksanddeeplearning.com, accessed Sep. 22. 2017. .
[26] Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. Girshick, S. Guadarrama, and T. Darrell, "Caffe: Convolutional architecture for fast feature embedding," Proceedings of the 22nd ACM international conference on Multimedia, pp.675–678, 2014.