WASHINGTON UNIVERSITY IN ST. LOUIS

School of Engineering and Applied Science

Department of Computer Science and Engineering

Thesis Examination Committee:
Dr. Ron Cytron
Dr. Tao Ju
Dr. Robert Pless

# USING REGRESSION TECHNIQUES TO PREDICT WEATHER SIGNALS FROM IMAGE SEQUENCES

by

Richard Speyer

A thesis presented to the School of Engineering
of Washington University in partial fulfillment of the
requirements for the degree of

MASTER OF SCIENCE

May 2009
Saint Louis, Missouri

ABSTRACT OF THE THESIS

Using Regression Techniques to Predict Weather Signals from Image Sequences

by

Richard Speyer

Master of Science in Computer Science

Washington University in St. Louis, 2009

Research Advisor: Dr. Robert Pless

Webcams are cheap sensors that capture a potentially large amount of information about a scene. This thesis considers the use of regression and correlation techniques such as Canonical Correlation Analysis (CCA) to convert these webcams into environmental sensors and predict the values of weather signals. Local environmental properties often directly affect the images we collect from the webcams; whether it is cloudy or sunny is visible by the presence of shadows; wind speed and direction is visible in smoke, flags, or close up views of trees; particulate density is reflected in haziness and the color spectrum during sunset. Using the AMOS database, which has been archiving nearly 1,000 webcams every 30 minutes for the last 3 years, we explore relationships between the amount of training data and the accuracy with which we are able to infer the values of certain weather signals including wind velocity and vapor pressure from inherent properties in the image. This allows the webcams *already* installed across the earth to act as generic sensors to improve our understanding of local weather patterns and variations.

# Acknowledgments

First and foremost, I would like to thank Dr. Robert Pless, who has been my research and faculty advisor for the last four years. His constant attention and guidance have been crucial in the construction of this thesis and I am extremely grateful for all of his help. My experience working under him in the Media & Machines Lab has been one of the most valuable of my college career and has taught me skills that I will surely use after I graduate.

I would also like to thank Nathan Jacobs and many other students who helped review this thesis and the related research as well as Dr. Tao Ju and Dr. Ron Cytron for sitting on my Thesis Committee along with Dr. Pless.

Finally, I would like to thank the Department of Computer Science & Engineering for promoting and providing opportunities for undergraduates to get involved in research projects. I have benefited greatly from working in a research lab during my time here and that would not have been possible without the department's commitment to undergraduate research.

Richard Speyer

*Washington University in Saint Louis*
*May 2009*

# Contents

# List of Figures

# Chapter 1

# Introduction

The appearance of a scene can often provide an abundance of information about the scene including time of day, location, and weather. With all of this information available, it is up to us to find ways to automatically extract it from images of the scene to allow a better understanding of what is going on at a given location. This thesis will attempt to use the information collected by webcams to infer information about environmental signals such as wind velocity and vapor pressure.

One of the primary challenges of predicting weather signals from images is that the images may vary more due to other causes such as time of day than those which relate to our signals of interest. Thus, we begin by exploring mechanisms for learning features that are invariant to other scene changes. Through the use of correlation techniques such as Canonical Correlation Analysis (CCA), we find that we are able to extract useful information from webcam images which allow us to predict local weather data and, as a result, gain a better understanding of local weather patterns and variations and fill in missing weather data entries, which occur quite frequently. Furthermore, this allows us to use the abundantly existing webcams all over the country as crude weather sensors instead of depending only on the government weather stations spread out sparsely across the country.

This thesis will go through the background information, motivation, and related work in Section 1. It will then discuss the theory and details of the correlation techniques used in Section 2. Section 3 will begin to show the application and results of the aforementioned methods. It will also show how we further utilize our results to determine the appropriate size of the training set necessary to avoid any biases in the images or over training of the predictors. We will conclude in Section 4 and propose future work on this problem in Section 5.

## 1.1 Related Work

The work presented in this paper is primarily related to two areas of research in computer vision; here we will present some of the work related to algorithms designed to operate on webcam image sequences as well as the use of CCA and other correlation techniques to extract external signals from time-varying image sequences.

**TODO**

## 1.2 Background Information

The Archive of Many Outdoor Scenes (AMOS) database [5] has been collecting images from 835 webcams every 30 minutes since March 2006 and now contains over 40 million images. The AMOS dataset is unique in providing time-stamped images from many cameras around the world. No other dataset provides the broad range of geographic locations and the long temporal duration that is does. This database is the largest known collection of natural scenes collected from static cameras and as such offers a wealth of data to test our methods against. While there are cameras located across the world, we focus on those located within the continental United States so that ground truth weather data can be collected.

The weather data is collected from the Historical Weather Data Archives (HWDA) [10] which is maintained by the National Oceanic and Atmospheric Administration (NOAA), an official government organization. The archives maintain a large variety of weather data from January 1, 1933 through present day on just over 6,000 weather stations located across the continental United States. The data collected includes, but is not limited to, wind velocity, precipitation, temperature, relative humidity, vapor pressure, cloud conditions, dew points, and various aggregated data signals.

# Chapter 2

# Canonical Correlation Analysis (CCA)

The main correlation technique which we will explore is a method called Canonical Correlation Analysis (CCA) [3]. The goal of CCA is to find two transformation matrices $A$ and $B$ to maximize the correlation between two independent data sets $X \in \mathbf{R}^{x \times n}$ and $Y \in \mathbf{R}^{y \times n}$. In other words, CCA looks to find $A$ and $B$ such that $AX \approx BY$. CCA is a way of finding a linear relationship between two independent, multidimensional variables.

**MORE, EQUATIONS ETC, EXPLANATION OF MULTIPLE DIMENSIONS**

What makes CCA different from other correlation methods is its invariance to affine transformations of the input variables. In other words, CCA will be able to find a linear relationship between two multidimensional variables even if different coordinate systems are used for each variable [2]. This makes CCA a very desirable method for relating image data to environmental signals, which are clearly not measured in the same coordinate system. One very crucial component of CCA is that while the input matrices $X$ and $Y$ can have a varying number of dimensions, they must have *exactly* the same number of samples in order to find a CCA relationship between two multivariable signals.

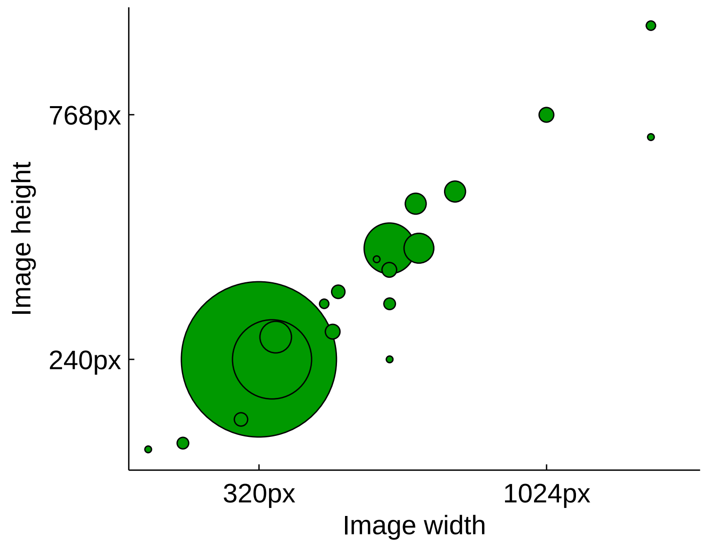



Figure 2.1: The distribution of image sizes, measured in pixels. The circles are centered at the width and height of the image and their sizes are proportional to the number of webcams which output images of that size in the AMOS dataset.

## 2.1 Applications

We will now focus on the application of CCA to predicting time-varying weather signals from image sequences over the same time period. Given the localized nature of weather data, we assume that the weather station used for ground truth weather data is located near to the camera in order to maximize the accuracy of our predictions. The algorithm takes as input a set of images $I = i_1 \ldots i_a$ and a set of weather observations $W = w_1 \ldots w_b$. The method will assume the availability of images and weather data with corresponding timestamps. In order to ensure that this invariant holds, the first step of the algorithm is to run through both datasets and remove entries that do not have a corresponding entry in the other dataset. This step guarantees that all of the data samples match and that there are an equal number in both sets, which is required for CCA. Now, the input is of the form $I = i_1 \ldots i_n$ and $W = w_1 \ldots w_n$.

Once the datasets are properly aligned, we turn our attention to the images. Figure 2.1 shows us that the most common size image from a webcam in the AMOS dataset is $320 \times 240$ pixels, which means one image can be expressed as a $1 \times 76800$ vector. This is clearly a very costly and inefficient way to store image data, especially when CCA will require a few hundred images to build a good predictor. In order to reduce the storage size for each image, and as a result accelerate the runtime of our algorithms, Principal Component Analysis (PCA) will be applied to the images as a way to find the $k$ most important features from the images and then express each image as a linear combination of these features ($k$=10 is used here).

PCA will take as input a set of images $I$ and will return three matrices $U \in \mathbf{R}^{m\times k}$, $S \in \mathbf{R}^{k\times k}$, and $V \in \mathbf{R}^{k\times n}$ where $m$ is the length of a single image when expressed as a vector. $U$ contains the $k$ orthogonal feature vectors, $S$ is a diagonal matrix where the elements represent the relative importance of each of the $k$ features, and $V$ contains the coefficients of each of the $k$ feature vectors for each of the $n$ images ($v_i$ contains the coefficients for the $i$th image). We can thus create reconstructions of the images by multiplying $U$, $S$, and $V$ back together. In mathematical terms, PCA attempts to find $U$, which is made up of $k$ orthogonal feature vectors such that we minimize:

$$\sum_{i=1}^{n} (I_i - USv_i)^2 \tag{2.1}$$

PCA will extract significant scene variations from the set of input images which, when used to reconstruct the original images, will minimize the reconstruction error. We now have a way to express each image as a $1 \times k$ vector, which is significantly smaller than the original image and contains the most important variations. This will be part of the input into CCA.

The PCA coefficients for each image stored in $V$ and the corresponding weather data $W \in \mathbf{R}^{y\times n}$ will be the input matrices for CCA. After running CCA, we will have projection matrices $A$ and $B$. These can now be used to predict weather data from new images which were not included in the input for CCA. Given a new image $i$, we begin by obtaining the $k$ PCA coefficients by projecting the image onto our existing basis vectors stored in $U$. We can now take those coefficients in a vector $\mathbf{v}$ and use them to predict the associated weather values $\mathbf{w}$ as follows:

$$\mathbf{w} = A\mathbf{v}B^{-1} \tag{2.2}$$

This is the key equation that is used to extract the inherent weather data from an image. We will now begin to apply these algorithms to actual data sets and present some results as well as additional information which can be extracted, including the minimum size of the training set and the orientation of the camera.

# Chapter 3

# Results & Analysis

Two weather signals are considered as driving examples: wind velocity and vapor pressure. These two signals present unique challenges and opportunities. The effect of wind velocity is limited to locations in the scene that are affected by wind, such as flags and vegetation. On the other hand, vapor pressure may affect the scene in a more broad and subtle manner. Choosing two examples with such unique characteristics is a great way to test whether the algorithm is able to handle a variety of weather signals or if it is best suited for certain classes of measurements.

## 3.1 Wind Velocity

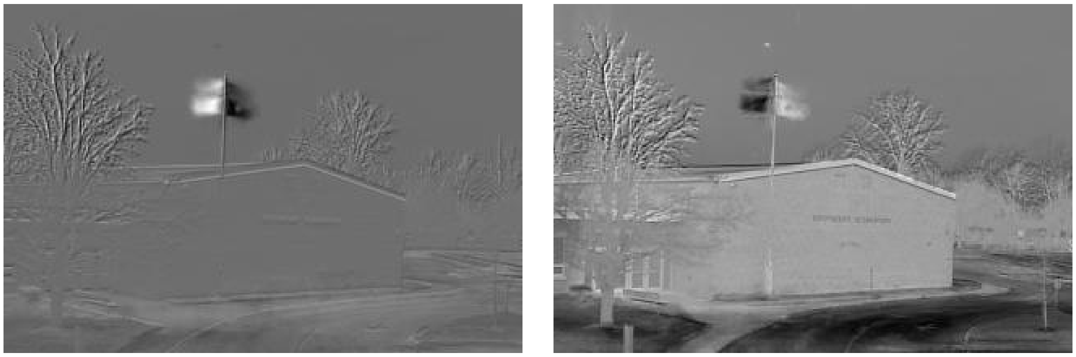
Wind velocity is a signal whose effects are only seen in certain local parts of the image; objects such as building and cars will be unaffected by changes in wind. In order to ensure that wind can be accurately predicted, a camera is chosen which contains a flag (Figure 3.1).

The CCA projections are trained on 204 images collected between January 1 and February 11, 2009. In order to focus on variations in the scene due to weather, we only use images captured between 10 AM and 2 PM local time. Doing this can successfully remove most of the variation caused by time of day and the resulting shadows. The wind velocity is made up of a wind speed in meters/second and a direction in degrees, which we convert to north/south and east/west components using basic trigonometry, and was collected from the closest weather station to the camera.


Figure 3.1: Some sample images from camera #194 of the AMOS database located in Decatur, IN. The presence of a flag on top of the school building is key to the ability to predict wind velocity.

Based on this data, we would expect the two dimensions of the CCA projection matrix $A$ to predict the north/south and east/west components of the wind speed. After running CCA, the matrix $A$ is projected onto the feature vectors $U$ from the PCA analysis of the images in order to visualize the canonical features extracted by CCA. As we see in Figure 3.2(a), CCA clearly identifies the position of the flag as the crucial indicator of wind speed. Furthermore, we can also notice slight variations in the tree positions, which also are affected by wind speed. The projection of the second basis vector, which can be seen in Figure 3.2(b) is far less accurate and does not yield an accurate prediction of the other component of the wind speed. This indicates that wind velocity can only be accurately predicted along one of the two components. While this is not a favorable result, it is plausible and agrees with our assumptions; the flag captured in a 2D image can clearly only predict wind speed along one vector.

We can now use equation 2.2 to infer the magnitude of the wind velocity on 102 images collected between February 11 and March 17, 2009 which were not used in the training of the CCA. Figure 3.3 shows a plot of the ground truth values for wind speed as well as the predicted values using CCA. The associated images below the plot correspond to the filled markers and clearly indicate that the wind speed predictions not only agree with the ground truth values but also directly correlate

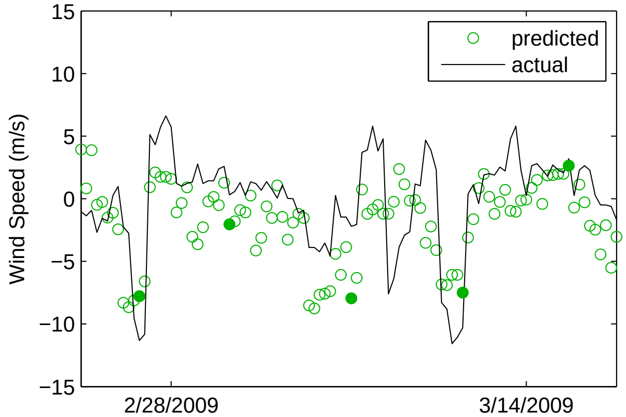


(a) (b)

Figure 3.2: Figure 3.2(a) shows the projection of the first basis vector from CCA onto the original image space. It is clear to see that the position of the flag and the trees are the major variations. Figure 3.2(b) shows the projection of the second basis vector and is far less accurate than the first.

with the direction of the flag. These results, along with the scatter plot in Figure 3.4 strongly support our belief that wind speed along a given vector can be predicted based on the position of the flag ($r = 0.61759$).

Although we are unable to predict both components of the wind speed, there is still some more information can be extracted from the results. Namely, we can use the known actual wind velocity vectors with our predicted wind speed along some unknown vector to solve for the direction of that vector. We do this by solving a simple matrix equation of the form $Ax = b$ where $A \in \mathbf{R}^{n\times 2}$ are the known wind speeds in north/south and east/west components and $b \in \mathbf{R}^{n\times 1}$ are the predicted wind speeds along an unknown vector which will be perpendicular to $x$. Thus, solving for $x$ will yield the vector perpendicular to the direction the flag blows in our camera. Figure 3.5 shows a scatter plot which visualizes the relationship between the actual and predicted wind speeds, along with the line solved for using the above equation. The same line is also shown overlaid onto an actual satellite image of the known camera location.




(a)

(b)

Figure 3.3: Predicted wind speed values and corresponding ground truth values in meters/seconds shown in Figure 3.3(a). Each image in Figure 3.3(b) is associated with one of the filled markers in the plot above.

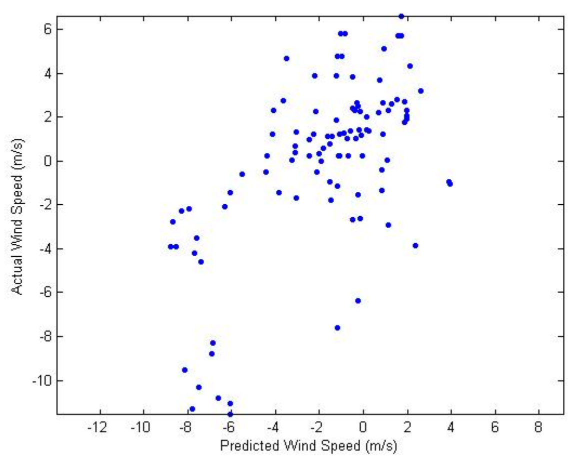
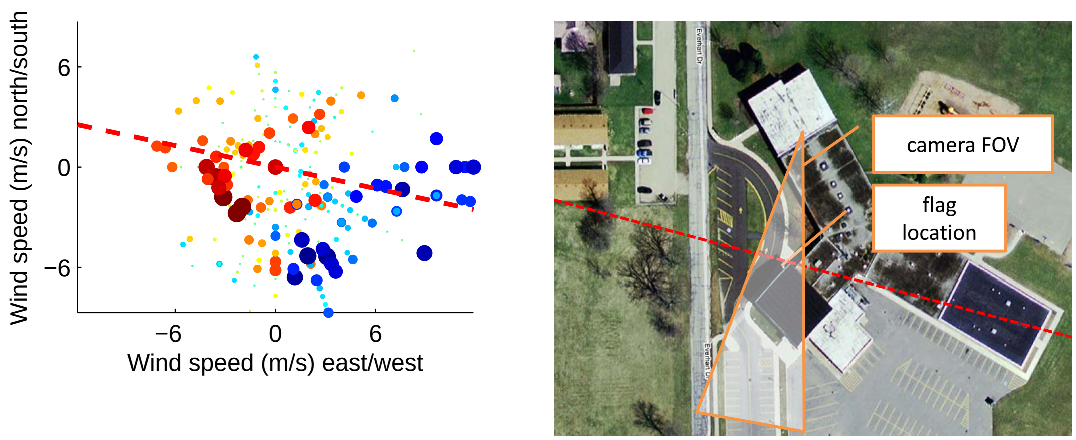



Figure 3.4: Scatter plot of predicted wind speed values vs. actual wind speed values. ($r = 0.61759$)




(a) (b)

Figure 3.5: Further analysis of the wind speed predictions provides a way to predict the axis along which the flag is blowing. In Figure 3.5(a) the size and color of each marker is determined by the predicted wind speed and the location of each marker is determined by the north/south and east/west components of the actual wind speed at the same time. The dashed red line is the normal to the projection axis determined by running linear regression between the predict and actual values. Figure 3.5(b) shows this axis overlaid on a Google Maps image with the field of view crudely estimated by hand.

Figure 3.6: Some sample images from camera #619 of the AMOS database located in Houston, TX. The visibility of the distant skyline of buildings as well as the presence of clouds to helps to predict vapor pressure.

## 3.2 Vapor Pressure

The second example will consider the weather signal of vapor pressure, which is the contribution of water vapor to the overall atmospheric pressure and is measured in millibars. Since this is a 1-dimensional signal, unlike wind velocity, running CCA is essentially equivalent to linear regression. A camera is chosen which contains a distant skyline of buildings as well as a view of the sky and horizon (Figure 3.6). The hypothesis is that as vapor pressure increases, the overall clarity of the distant skyline will decrease due to haze and clouds. The primary challenge with vapor pressure as opposed to wind speed is that it is a signal which effects the entire scene as opposed to a localized area, and is not quite as easy to comprehend visually.

The CCA projections for this example were trained on 198 images captured from January 1 to February 19, 2009. Once again, we only consider images between 10 AM and 2 PM as a way to ignore variations due to time of day.

Unfortunately, when the algorithm is run exactly as described above, we get poor results that do not correlate very strongly with what we expect to change with vapor pressure. As is seen in Figure 3.7(a), the reprojection of the single dimension of CCA does not yield a very convincing image. It appears that the CCA projection has

(a)

(b)

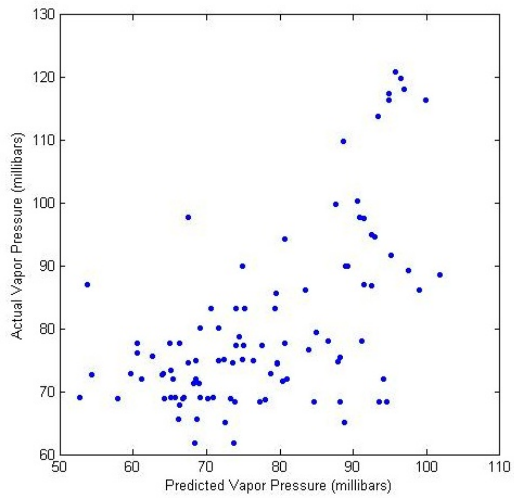
Figure 3.7: The projections of the CCA basis vector computed with the regular and gradient images, respectively. Figure 3.7(a) does not identify a feature which actually correlates strongly with vapor pressure. However, when using the gradient image instead in Figure 3.7(b), the CCA basis clearly identifies the visibility of the skyline as an indicator of vapor pressure, which agrees with the hypothesis.

identified the position of the sun against the buildings as one of the important factors, which does not have anything to do with vapor pressure. Despite CCA identifying this seemingly unrelated signal, the predictions yield a decent correlation ($r = 0.59968$) between themselves and the actual values, as seen in Figure 3.8(a). In order to improve the predictions and identify a relevant feature in the image, the original images are replaced with their gradient magnitude images, which are computed as follows:

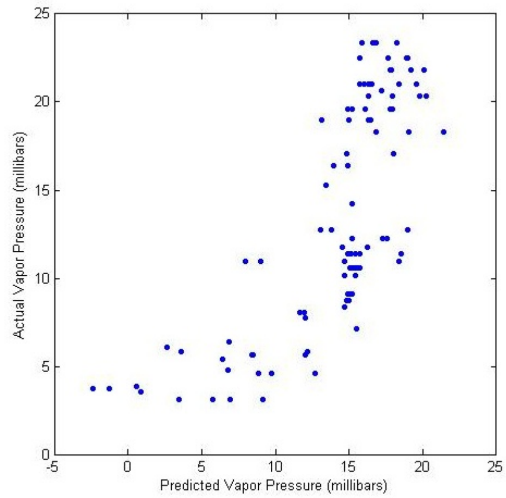
$$I_g = \sqrt{\frac{dI}{dx}^2 + \frac{dI}{dy}^2} \tag{3.1}$$

The gradient images will tend to highlight edges and other major changes in the images. The projection of the CCA basis found when using these gradient images can is in Figure 3.7(b) and, as we can see, clearly focuses on the visibility of the edges of the buildings, which is what was hypothesized to change with vapor pressure. Additionally, we get a stronger correlation between the actual and predicted values when using the gradient image ($r = 0.73684$).

Further analysis of our results indicate that vapor pressure is being correctly predicted instead of some other signal which happens to correlate strongly with vapor pressure. Figure 3.9 shows the time series predicted vapor pressure data along with the ground truth values. We see that high vapor pressure measurements clearly indicate more




(a)



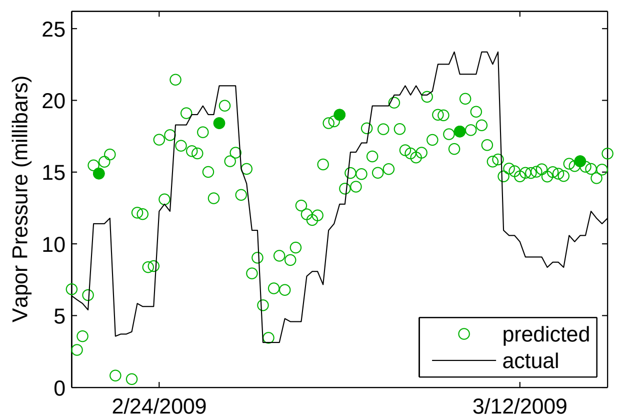

(b)

Figure 3.8: Scatter plots of the predicted vs. actual vapor pressures (in millibars) computed with the regular and gradient images, respectively. While the correlation with the regular images (Figure 3.8(a), $r = 0.59968$) is not poor, there is a noticeable improvement when using the gradient images (Figure 3.8(b), $r = 0.73684$).

cloudy days, which is exactly what was expected. This example also identifies one failure mode of our algorithm. On March 12, there was such a heavy fog that it obscured the entire scene, making it impossible to accurately predict vapor pressure. This is an unavoidable failure mode as the algorithm depends on the appearance of the image and nothing else to predict the weather. Despite this failure mode, it is clear that the algorithm is able to successfully predict vapor pressure given a reasonable image of the scene.

## 3.3 Error Analysis

A standard question that must be asked of any machine learning algorithm is: how large of a training dataset is necessary to build a strong model? A strong model is one that can predict our given weather data both accurately *and* precisely. That is, predictions from novel data have small error residuals and are not biased in one direction or the other. The combination of accuracy and precision is a strong argument that the model is truly predicting weather data and not some other signal from the




(a)

(b)

Figure 3.9: Predicted wind speed values and corresponding ground truth values in millibars shown in Figure 3.9(a). Each image in Figure 3.9(b) is associated with one of the filled markers in the plot above. The poor predictions on March 12 are due to heavy fog which obscured the entire scene and left water on the optics.

Figure 3.10

image that is similar to the weather. The goal now is to find the minimum number of data samples needed to build such a model.

Recall the wind velocity examples presented in Section 3. We used 204 images to train the CCA projections and tested the results on 102 images. We will now analyze the performance of our algorithms on training sets of various sizes. The images will be selected in a contiguous block of days and will still be limited to those collected between 10 AM and 2 PM. As is seen in Figure 3.10, we see that the rate at which the correlation coefficient $r$ improves stops changing drastically around 140 images.

# Chapter 4

# Conclusion

# Chapter 5

# Future Work

One of the things which makes the AMOS dataset so special and unique is the extremely large number of cameras which is contains. However, only a very few of them are being taken advantage of in this thesis. The algorithm presented in this thesis has been shown to behave well on cameras whose scenes are very different and weather signals that vary as well.

The goal of any future work on this subject should be to further automate the process of finding weather stations close to AMOS cameras, downloading the available weather data, and constantly running, and updating, CCA to predict a variety of weather signals on many cameras. This is very valuable work as it will allow us to analyze local weather patterns across multiple cameras and to infer the weather at other cameras using its location and the locations of other cameras around it. This will allow us to make weather predictions at locations that do not necessarily have a weather station within a reasonable vicinity.

# Appendix A

# Camera Information

# Appendix B

# Weather Data

# References


[1] Robert S. Barcikowski and James P. Stevens. A monte carlo study of the stability of canonical correlations, canonical weights and canonical variate-variable correlations. In *Multivariate Behavioral Research*, volume 10:3, pages 353–364, 1975.

[2] M. Borga, H. Knutsson, and T. Landelius. Learning canonical correlations. In *SCIA10*, Lappeenranta, Finland, June 1997. SCIA.

[3] H. Hotelling. Relations between two sets of variates. In *Biometrika*, volume 28, pages 321–377, 1936.

[4] Nathan Jacobs, Walker Burgin, Richard Speyer, David Ross, and Robert Pless. Adventures in archiving and using three years of webcam images. In *IEEE Workshop on Internet Vision (CVPR 2009)*, Miami, FL, June 2009. (to appear).

[5] Nathan Jacobs, Nathaniel Roman, and Robert Pless. Consistent temporal variations in many outdoor scenes. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 2007)*, Minneapolis, MN, June 2007.

[6] Nathan Jacobs, Nathaniel Roman, and Robert Pless. Toward fully automatic geo-location and geo-orientation of static outdoor cameras. In *IEEE Workshop on Applications of Computer Vision (WACV 2008)*, Copper Mountain, CO, January 2008.

[7] Nathan Jacobs, Scott Satkin, Nathaniel Roman, Richard Speyer, and Robert Pless. Geolocating static cameras. In *Proc. IEEE International Conference on Computer Vision (ICCV)*, Rio De Janiero, Brazil, October 2007.

[8] Dongge Li, Nevenka Dimitrova, Mingkun Li, and Ishwar K. Sethi. Multimedia content processing through cross-modal association. In *MULTIMEDIA '03: Proceedings of the eleventh ACM international conference on Multimedia*, pages 604–611, New York, NY, USA, 2003. ACM.

[9] S.G. Narasimhan, C. Wang, and S.K. Nayar. All the Images of an Outdoor Scene. In *Europian Conference on Computer Vision (ECCV)*, volume III, pages 148–162, May 2002.

[10] National Oceanic and Atmospheric Administration. Historical weather data archives, April 2009. http://data.nssl.noaa.gov/.

# Vita

Richard Speyer

| | |
|---|---|
| **Date of Birth** | October 11, 1986 |
| **Place of Birth** | Johannesburg, South Africa |
| **Degrees** | B.S. Computer Science, May 2009<br>M.S. Computer Science, May 2009 |
| **Professional Societies** | Association for Computing Machinery |
| **Publications** | N. Jacobs, S. Satkin, N. Roman, R. Speyer, and R. Pless. Geolocating static cameras. In *Proc. IEEE International Conference on Computer Vision*, Oct. 2007.<br><br>N. Jacobs, W. Burgin, R. Speyer, D. Ross, and R. Pless. Adventures in Archiving and Using Three Years of Webcam Images. In *IEEE Workshop on Internet Vision (CVPR 2009)*, Jun. 2009. (to appear) |

May 2009

**Predicting Weather from Image Sequences, Speyer, M.S. 2009**