UNIVERSITY OF CALIFORNIA
MERCED

# Geographic and Environmental Interpretation of Photographs

A Thesis submitted in partial satisfaction
of the requirements for the degree of

Master of Science

in

Electrical Engineering and Computer Science

by

Ling Xie

Committee in Charge:

Prof. Shawn Newsam[1], Chair

Prof. Ming-Hsuan Yang[1]

Prof. Qinghua Guo[2]

June 2011

[1]Electrical Engineering and Computer Science Graduate Group, University of California, Merced
[2]Environmental Systems Graduate Group, University of California, Merced

The Thesis of
Ling Xie is approved:

---

Prof. Ming-Hsuan Yang[1]

---

Prof. Qinghua Guo[2]

---

Prof. Shawn Newsam[1], Committee Chairperson

June 2011

Geographic and Environmental Interpretation of Photographs



by

Ling Xie

# Acknowledgements

I would like to express my appreciation to my advisory committee: Prof. Shawn Newsam, Prof. Ming-Hsuan Yang and Prof. Qinghua Guo. Thanks for giving me the opportunity to be part of the geographic vision research. Special thanks to Prof. Newsam for his time, patience, and understanding. It has been an honor to work with you. Also, thanks to UC TSR&TP lead campus training grant for graduate student research and UC CITRIS Seed Funding, which supported me early in this research. My gratitude also goes to the Computer Vision Lab at UC Merced, there are not enough words to describe my appreciation. Prof. Ming-Hsuan Yang and Qinghua Guo, thanks for the help with the discussions and suggestions, which have helped my greatly.

Finally, thanks to all the people who gave me help during my study in UC Merced. I cannot imagine how tough the life here would be without your help.

# Abstract

## Geographic and Environmental Interpretation of Photographs

Ling Xie

The geographic and environmental interpretation of photographs is of increasing interest due to the growing availability of large scale datasets annotated with location information. An automatic interpretation system can assist humans in undertanding landscapes of large areas, provide cues for the scenicness of a location, or pre-filter the unrelated images for further processing, such as object and event recognition. My work mainly focuses on two problems: estimating atmospheric visibility from static images and estimating the "scenicness" of the scene depicted in an image and generating a corresponding map.

For the problem of using images to estimate atmospheric visibility, I construct the analytic mapping from image feature space, which consists of spatial contrast between pixels or frequency energy, to the optical measure of atmospheric visibility. Also, I explore a manifold learning algorithm from machine learning to obtain a more reliable estimator by constructing a graph of images based on their temporal order, since the direct mapping is very challenging.

To automatically estimate the scenicness of an image, I perform experiments using global image features. A set of manually labeled images are used to learn a regressor for predicting the labels of novel images. I also investigate two extensions: first, I propose a novel composite visual-geographic location kernel which considers both the visual sim-

ilarity and geographic proximaty of images; and second, I improve the accuracy of the learned regressor by incorporating unlabeled images in a graph Laplacian semi-supervised learning framework. I also investigate a novel composite visual-geographic graph Laplacian regularization term. I demonstrate the approach by using ground-level images to label how scenic a location is.

---

Prof. Shawn Newsam[1]
Dissertation Committee Chair

# Contents

# List of Figures

# List of Tables

# Chapter 1

# Introduction

This work focuses on the geographic and environmental interpretation of photographs. The Merriam-Webster dictionary defines the terms geography as "a science that deals with the description, distribution, and interaction of the diverse physical, biological, and cultural features of the earth's surface", and environment as "the complex of physical, chemical, and biotic factors (as climate, soil, and living things) that act upon an organism or an ecological community and ultimately determine its form and survival".

My work investigates automatic means for extracting geographic and environmental knowledge from large collections of readily available, non-specialized photographs. In particular, I focus on the following two problems:

1. Estimating atmospheric visibility from large collections of images from static cameras;

2. Estimating the "scenicness" of the scene depicted in an image;

The approaches I investigate for solving these problems can be divided into:

1. Visual feature selection and comparison to explore their correlation with scientific measures of atmospheric visibility.

2. Exploiting the temporal distribution of the photos to improve the visibility estimation.

3. Exploiting the spatial distribution of images to improve the scenicness estimation.

# Chapter 2

# Motivation

The geographic and environmental interpretation of photographs is of increasing interest due to the growing availability of large scale datasets annotated with location information. For example, the online photo sharing website, Flickr, has over 100 million geo-located photos, of which 2.6 million were added in the last month. An automatic interpretation system can assist humans in undertanding landscapes of large areas, provide cues for scene rating, or pre-filter images for further processing, such as object and event recognition.

## 2.1 Estimating atmospheric visibility

Due to their low cost, general-purpose cameras are being used to monitor a variety of phenomenon. Examples include surveillance, real-time traffic reporting, tracking the progress of construction projects, entertainment, and monitoring environmental conditions, such as the weather. My work focuses on an important application from this last category, that of estimating atmospheric visibility. In particular, I investigate the use of

image features automatically extracted from digital cameras as surrogates for traditional measures of atmospheric visibility from transmissometers.

Even if general-purpose cameras are not able to measure visibility as accurately as specialized equipment, they represent an attractive alternate since they are considerably less expensive and easier to deploy (a transmissometer requires a precisely calibrated transmitter and receiver separated by several kilometers and costs over $10,000).

The images acquired from visibility cameras are currently used for qualitative analysis only. I explore automated techniques for deriving quantitative measures of visibility based on image contrast and acuity. A ground truth dataset is used to assess the effects of parameter settings. My findings are an important step towards using low-cost general-purpose cameras for quantitative visibility monitoring.

## 2.2 Scene recognition

Scene recognition is an important image understanding problem. The scene is defined as the physical setting of the environment where the image is taken, such as outdoor, indoor, beach, mountain, forest, office or urban landscape [1]. Scene interpretation has a wide range of applications. Scene recognition can provide a general understanding of the environment where the image was taken. It can also be utilized as pre-filter or context prior for further recognition tasks such as object recognition. However, scene interpretation remains a significant challenge to the computer vision research community due to interclass variability, illumination effects and scale changes.

In general, a scene is modeled hierarchically. A traditional modeling approach is, on the lower level, scene images are modeled as the statistical distribution of pixels, or regions of interest, and on the upper level, a scene is modeled as the incorporation of different objects. But, recently, scene images have been modeled at the higher level as the compostion of intermediate *semantics*. For instance, in [2], manually labeled semantic concepts, e.g openness and naturalness, are analyzed as cues to understanding natural scenes.

My work performs geographic discovery, specifically scenicness estimation, in large collections of georeferenced community contributed photo collections but differs from previous work in two significant ways: 1) I use a combined visual-geographic location kernel to learn and apply a regression model, and 2) I use semi-supervised learning based on a combined visual-geographic graph Laplacian.

# Chapter 3

# Related Work

## 3.1 Atmospheric Visibility

While there is a large body of work on the related problem of improving the fidelity of images taken under hazy conditions, not much effort has investigated using the images to measure atmospheric visibility. Caimi et al. [3] review the theoretical foundations of estimating visibility using image features such as contrast, and describe a Digital Camera Visibility Sensor system, but they do not apply their technique to real data. Kim and Kim [4] investigate the correlation between hue, saturation, and intensity, and visual range in traditional slide photographs. They conclude that atmospheric haze does not significantly affect the hue of the sky but strongly affects the saturation of the sky, but they do not use the image features for estimating visibility. Baumer et al. [5] use an image gradient based approach to estimate visual range using digital cameras but their technique requires the automatic detection of a large number of targets, some only a few pixels in size. This detection step is sensitive to parameter settings and would not be robust to camera movement. Also, for ranges over 10 km, they only compare their

estimates with human observations which have limited granularity. Luo et al. [6] use Fourier analysis as well as image gradient to estimate visibility but they also only compare their estimates with human observations. Raina et al. [7] do compare their estimates with measurements taken using a transmissometer-like device but their approach requires the manual extraction of visual targets. The work by Molenar et al. [8] is closest to my technique in that it is fully automated and the results are compared with transmissometer readings. However, their technique uses a single distant, and thus small, mountain peak to estimate contrast and thus is very sensitive to camera movement.

A related but different problem is that of using visibility analysis for extracting geometric (such as depth) and radiometric (such as color) information for a scene. An optical scattering model is proposed by Shree et al. in [9], which views the process of information carried from the scene point to observer as a mechanism of visual information coding, and decomposes the signal arriving at the optical receiver into airlight and attenuation components, both of which contain information about spatial distance and air condition. They also propose a *dichromatic* atmospheric scattering model describing the dependence of atmospheric scattering on wavelength. But this model requires a clear day image for calibration, which is sometimes hard to collect. In [10], they introduce a general chromatic framework to extract information from images taken under different poor weather conditions. This framework derives geometric constraints on scene color changes and uses those constraints to recover scene colors as appearing on a clear day, in order to avoid the requirement of a clear day image. However this approach needs the weather conditions to change between image acquisitions, which can take too long to make dehazing practical,

and it also needs the scattering to be invariant to wavelength. In [11] and [12], they apply polarization filters and physics theory respectively to relax the constraints on the scattering model.

Another work which investigates temporal variation in image sequences is [13]. Here, PCA is used to extract the most significant variations for further image annotation and recognition.

## 3.2 Scene Recognition

Scene interpretation has attracted the attention of the computer vision community for a long time. Indoor/outdoor scene classification is one simple example of scene interpretation. Early work on indoor/outdoor classification was done by Szummer and Picard in [14] on the MIT indoor/outdoor dataset which consists of a collection of 1343 consumer photographs. Color histogram and texture features are used to discriminate between indoor and outdoor scenes through K-nearest neighbor classification. Similarly, the work in [15] and [16] computes the color histogram on both the Ohta and LST color spaces. But in their experiment the intersection distance is shown to outperform the Euclidean distance for nearest neighbor classification. In [17], the first order statistics of color and grayscale histograms are used as features in classification. Wavelet texture features based on a two level decomposition used in [15] and [16] are shown to be better than other forms of texture features. In [18] local dominant orientation (LDO) is used to represent the whole image. The papers of [19] and [20] claim indoor images have more

pronounced straight edges than outdoor scenes, so that shape description of the texture of edges is utilized.

Further study on more complex scene classification problems has been done in recent years, many of which use a common evalution dataset composed of images of 15 scene classes[1]. Oliva and Torralba [21],[22] extend the idea in [14] to incorporate global frequency, i.e multiscale oriented Garbor filters, and local constraints together. An intermediate representation is extracted from global and local frequency features for further scene classification. Li Fei-Fei et al. [23] adapt a *latent Dirichlet allocation* model (LDA) from text classification to scene recognition, where the concept of *topic* is replaced by *objects* possibly appearing in images, and develop a series of generative models to solve scene and object detection and classification tasks. However, the work in [23] ignores the spatial information, and treats the low-level features as bags of visual words. Sudderth et al. [24] model the spatial locations of detected features by associating objects with distributions over parts, which could share features through both LDA and constellation models. Similarly, in [25], the authors extend a hierarchical Dirichlet process to a nonparametric model, and use a tree structured graphical model to couple the visual topic assignments at nearby positions and scales. Another latent probabilistic model exploited in text recognition, probabilistic Latent Semantic Analysis(pLSA), is also used for scene classification [26],[27],[28]. But different from the implementation of LDA, the learned semantic features from pLSA, are used as input to a discriminative model, e.g

[1] `http://www-cvr.ai.uiuc.edu/ponce_grp/data/`
The 15 classes are: bedroom, suburb, industrial, kitchen, living room, coast, forest, highway, inside city, mountain, open country, street, tall building, office and store

SVM, to complete the classification task. In [1], a co-clustering via maximization of mutual information (MMI) based intermediate concepts discovering algorithm is applied in scene classification which, unlike pLSA, does rely on hidden variables and independence assumptions between images and visual words, given the latent variables.

Discriminative models have also shown to be effective for scene classification. Spatial pyramid matching (SPM) [29] is a spatial extension of the pyramid matching kernel to pre-compute the kernel of images in a training set using visual words. Unlike a generative model, SPM is applied directly to the visual words to measure the similarity between images. The work in [30] uses the kernel trick for codebook matching and a Gaussian kernel to make assignments of visual words instead of k-nearest neighbor. Jianchao Yang et al. [31] use sparse coding for the codebook assignments, and achieve much better results. Instead of using traditional unsupervised clustering, Jianxin Wu et al. [32] employ a histogram intersection kernel to learn the codebook. Most recently, Shenghua Gao et al [33] adapt sparse coding based on a local Laplacian graph to feature quantization, and achieve close to state of the art results.

## 3.3 Georeferenced Image Collections

The growing collections of georeferenced community contributed photo collections such as available at Flickr and Panoramio have attracted increasing attention from the computer vision research community. One particularly interesting and novel application of these collections is for geographic discovery. Computer vision and other knowledge

discovery methods have been applied to the photo collections to discover what-is-where on the surface of the earth in the broad sense of the term "what". Examples of work in this area include using large collections of georeferenced images to discover spatially varying (visual) cultural differences among concepts such as "wedding cake" [34]; to discover interesting properties about popular cities and landmarks such as the most photographed locations [35]; to estimate weather satellite images using widely distributed Web cameras [36]; to create a map-like partitioning of a country-sized region into geographically coherent subregions [37]; and to create maps of developed and undeveloped regions [38].

# Chapter 4

# Estimating Atmospheric Visibility

*Results published in the International Symposium on Visual Computing, 2008* [39]

## 4.1 Data Access and Dataset Generation

The dataset consists of images and transmissometer measurements from the Phoenix region acquired as part of the PhoenixVis.net project managed by the Arizona Department of Environmental Quality. Digital images with resolution 1600x1200 pixels were acquired every 15 minutes for two scenes, Camelback Mountain (CAME) and South Mountain (SOMT). Sample images are shown in figure 4.1. The SOMT images span all of 2006 while the CAME images span January through mid-August. We select only images taken at noon to minimize the effect of sun angle.

Since contrast and frequency features are based on pixel differences, we do not expect them to depend on small variations in the overall brightness between images. Thus, we do not account for differences in camera settings such as from auto-exposure since the images are acquired at the same time each day.


**Figure 4.1:** Sample image of Camel Mountain (left: CAME) and South Mountain (right: SOMT). Contrast is computed as the difference between bands of pixels above and below the horizon (left). Different sized bands are considered.

The ground truth transmissometer readings, $b_{ext}$, were acquired every hour for 2006. The cameras are located on mountains to the north and south of Phoenix and face each other. The transmissometer is located between the mountains. So, while the transmissometer is within the field of view of the cameras, spatial inhomogeneity in the atmosphere could limit the correlation between measurements from the cameras and the ground truth measurements from the transmissometer.

## 4.2 Methodology

Visibility is a measure of how well an observer can see through the atmosphere. This can either refer to the maximum distance at which a dark object is just visible against the background sky, also known as the visual range, or, more generally, as the clarity of objects in the distance, middle, or foreground. Our approach uses the later interpretation and employs both measures of contrast in the spatial domain and energy in the frequency domain as estimates of visibility. We compare our estimates to those of the

transmissometer which assesses visibility by measuring the amount of light lost over a known distance.

Transmissometers measure the extinction of light $b_{ext}$ which is related to observed contrast $C_r$ of an object viewed against the sky at a distance $r$ through [40]

$$\frac{C_r}{C_0} = \exp^{-b_{ext}r}. \tag{4.1}$$

Here, $C_0$ is the contrast of the object when $r = 0$; i.e., when there is no extinction by the intervening atmosphere.

### 4.2.1 Contrast in the spatial domain

We first measure observed contrast using the horizon. Assuming the location of the horizon is known, $C_r$ is computed as the difference between the means of the pixel values above and below the horizon. Let $P_{above}$ be the set of pixels above the horizon and $P_{below}$ be the set of pixels below the horizon. Then,

$$C_r = \frac{1}{\#P_{above}} \sum_{p \in P_{above}} f(p) - \frac{1}{\#P_{below}} \sum_{p \in P_{below}} f(p) \tag{4.2}$$

where $f(p)$ is the value of pixel $p$. For the sake of this work, we use a manually refined segmentation mask to determine the location of the horizon. This mask is derived once using a relatively clear day early in the year.

We refer to $C_r$ above as the absolute contrast. We also compute the relative contrast by dividing $C_r$ by the mean of the pixel values above the horizon. This is motivated by the fact that the human visual system's ability to discriminate between a target and

background depends not only on the difference between their intensities but also on the intensity of the background.

To investigate how localized the contrast measurement should be, we consider different sized bands of pixels above and below the horizon. A particular case for a band of size $d$ pixels is shown in figure 4.1 left. In order to make the approach more robust to moderate camera movement, we discard different sized margins of pixels right at the horizon ($d_{discard} << d$). Since the scattering and absorption of light by particles is wavelength dependent, we compute contrast using each of three color channels, red, green, and blue, as well as intensity (grayscale values). Thus, $f(p)$ in equation 4.2 above is the red, green, blue or grayscale value of pixel $p$.

Finally, we use image registration techniques to compensate for camera movement due to wind. We assume each image has undergone an affine transformation. The translation, rotation and scale parameters of this transformation are estimated using the method in [41] and the horizon segmentation mask is updated accordingly.

### 4.2.2 Energy in the frequency domain

We also consider the energy of the image within select frequency bands as a measure of visibility. The images are split into non-overlapping blocks to localize the analysis. The energy for each block is then computed as

$$Energy = \sum_{u,v} ||F_{block}(u,v) \cdot H_{u,v}||^2 \tag{4.3}$$

where $F_{block}(u, v)$ is the Discrete Fourier Transform (DFT) of a block, $H(u, v)$ is a frequency selective filter, and $u, v$ are the coordinates in the 2D discrete frequency domain. Three kinds of ideal filters are considered: low-pass, band-pass, and high-pass. A range of cut-off values is considered for each filter type. We perform a linear fit between the computed energy and the transmissometer readings.

The atmosphere is considered to be a low-pass filter [42] so we expect that the effect of the atmosphere will be greatest in image regions with high-frequency information. This makes intuitive sense since distant scenes appear less detailed on hazy days than clear days.

## 4.3 Results

The primary objective of this part is to use the ground truth transmissometer readings to compare the different approaches and parameter settings described in section 4.2. For each approach, we compute the correlation between our estimate of the extinction coefficient, $b'_{ext}$, and the ground truth as measured by the transmissometer, $b_{ext}$. In the spatial domain, $b'_{ext}$ is computed as the log of the contrast as computed in equation 4.2. In the frequency domain, $b'_{ext}$ is the energy as computed in equation 4.3. We perform (vertical) linear regression between the estimated and measured extinction coefficients, as shown in figure 4.2, and use the coefficient of determination, $R^2$, as a measure of the fit. A larger $R^2$ value indicates a higher correlation and thus a better approach. This allows us to compare the approaches without providing explicit values for $C_0$, $r$, or other

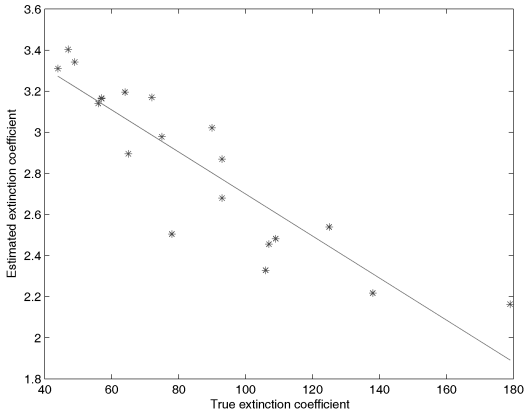



**Figure 4.2:** Example of linear regression between $b_{ext}$ measured using a transmissometer (the ground truth) and $b'_{ext}$ computed as the log of image contrast. Note we are only interested in the quality of the fit and not the actual values.

scaling and offset parameters, since they do not affect the quality of the fit as measured using $R^2$. If our objective was to predict $b_{ext}$ then we would need to determine the values of these scaling and offset parameters.

In order to minimize the effect of seasonal variation, we perform the linear regression one month at a time and use the average of the monthly $R^2$ values for comparing approaches.

### 4.3.1 Contrast in the spatial domain

We assume that the effects of the different approaches are to a first order approximation independent in order to avoid evaluating all possible combinations. The size of the band of pixels turns out to have the largest effect so we consider it first. Table 4.1 shows the $R^2$ values for different sized bands. The optimal setting is scene dependent.

**Table 4.1:** $R^2$ values for bands of different sizes.($d$ in figure 4.1)

| | **25** | **50** | **100** | **200** |
|---|---|---|---|---|
| **CAME** | 0.604 | 0.580 | 0.550 | 0.432 |
| **SOMT** | 0.182 | 0.389 | 0.604 | 0.524 |

A smaller band is shown to be better for CAME. We hypothesize this could be for two reasons: 1) it is taken from a lower vantage point than SOMT and thus the regions below the mountains are too close to be effective targets, and 2) it undergoes less camera movement. The results from incorporating image registration described below help to resolve this.

We next fix $d = 25$ for CAME and $d = 100$ for SOME, and investigate the effect of the size of the band of discarded pixels, $d_{discard}$. We originally postulated that ignoring pixels close to the horizon would make the contrast measurement more robust to camera movement. However, even though both cameras move over the course of the year, the difference in $R^2$ values between discarding 0, 5, and 10 pixels is less than 0.01 for both cameras. It seems that the effects of misalignment of the segmentation mask are averaged out by the size of the regions used to compute the contrast.

The scattering of light by small particles is wavelength dependent so we investigated which spectral channel is most effective for measuring contrast. Atmospheric scattering is most pronounced for blue light (this is why the sky is blue), leading us to postulate that there might not be sufficient contrast between the mountain and sky in the blue channel. The $R^2$ values for different spectral channels are shown in table 4.2. The results are mixed and scene dependent. The blue channel is shown to be the worst choice for

**Table 4.2:** $R^2$ values for different spectral channels.

| | **red** | **green** | **blue** | **intensity** |
|---|---|---|---|---|
| **CAME** | 0.520 | 0.627 | 0.617 | 0.604 |
| **SOMT** | 0.601 | 0.599 | 0.575 | 0.604 |

SOMT but not for CAME. The green channel is optimal or near-to for both cameras so it is the best channel overall. This finding agrees with previous results [8]. But, intensity performs comparable to the green channel so it is a reasonable choice since intensity images are smaller and thus less costly to transmit and store (assuming that the spectral separation cannot performed at the remote camera site).

Using absolute instead of relative contrast resulted in higher $R^2$ values for both cameras but the increase was only around 0.02. For completeness, we investigated a linear relationship between observed contrast and extinction coefficient. This resulted in lower $R^2$ values for both cameras confirming that the exponential relationship of equation 4.1 is correct.

The cameras are observed to move during the course of the year which can result in inaccuracies in the segmentation mask used to locate the horizon. Therefore, we apply image registration techniques [41] to estimate the camera movement between consecutive images and update the mask. This is shown to improve the results, especially for SOMT which experiences more movement. Table 4.3 shows the results for different sized bands when using registration to update the mask. Compare these results with those in table 4.1. A smaller band is now optimal for both cameras. The previously observed poor

**Table 4.3:** $R^2$ values for bands of different sizes when image registration is used to correct for camera movement.

| | **25** | **50** | **100** | **200** |
|---|---|---|---|---|
| **CAME** | 0.607 | 0.581 | 0.552 | 0.431 |
| **SOMT** | 0.617 | 0.598 | 0.489 | 0.608 |

performance of SOMT with a smaller band can now be attributed to the increased camera movement and not the difference in vantage point.

### 4.3.2 Energy in the frequency domain

Since the atmosphere can be considered a low-pass filter, we expect the high frequency information in an image to be most informative for estimating visibility. To confirm this, we compute the energy in different frequency ranges by filtering with low-pass, band-pass, and high-pass filters.

Since the frequency content of an image varies spatially, we first partition the image into 24 non-overlapping 256-by-256 pixel blocks. The energy of each filtered block is then computed using equation 4.3. We perform linear regression between the energy of each block and $b_{ext}$ as measured by the transmissometer to assess the correlation between different frequency components and visibility.

Table 4.4 presents the results of filtering the blocks using a low-pass filter with a cutoff frequency ranging from 0+, which corresponds to the DC component only, to 1.0, which corresponds to an all-pass filter. The results are the average $R^2$ values for the six horizontal blocks containing the horizon. These were found to be more informative

**Table 4.4:** $R^2$ values for low-pass filters with cutoff frequencies ranging from 0+, which corresponds to the DC component only, to 1.0, which corresponds to an all-pass filter.

| | **0+** | **0.2** | **0.4** | **0.6** | **0.8** | **1.0** |
|---|---|---|---|---|---|---|
| **CAME** | 0.114 | 0.382 | 0.466 | 0.490 | 0.498 | 0.501 |
| **SOMT** | 0.349 | 0.115 | 0.376 | 0.453 | 0.477 | 0.486 |

**Table 4.5:** $R^2$ values for high-pass filters with cutoff frequencies ranging from 0-, which corresponds to an all-pass filter except for DC, and 1.0-, which corresponds to a filter which preserves only those components outside the largest circle that can be inscribed inside the 2D discrete frequency plane (i.e., the corners).

| | **0-** | **0.2** | **0.4** | **0.6** | **0.8** | **1.0-** |
|---|---|---|---|---|---|---|
| **CAME** | 0.507 | 0.534 | 0.522 | 0.502 | 0.484 | 0.434 |
| **SOMT** | 0.499 | 0.560 | 0.553 | 0.541 | 0.531 | 0.530 |

than the other 256-by-256 blocks. As expected, the best results are obtained using the high-frequency information.

Table 4.5 presents the results of filtering the blocks using a high-pass filter with a cutoff frequency ranging from 0-, which corresponds to an all-pass filter except for DC, and 1.0-, which corresponds to a filter which preserves only those components outside the largest circle that can be inscribed inside the 2D discrete frequency plane (i.e., the corners). There are two interesting things to note about these results. First, including low frequency information degrades the performance. Second, the best results include the mid-range frequencies.

To further explore the mid-range frequencies, we applied a band-pass filter with a pass band centered at one half the Nyquist frequency, and a bandwidth ranging from 0+, which corresponds to a narrow pass band, to 1.0, which corresponds to an all-pass filter.

**Table 4.6:** $R^2$ values for band-pass filters centered at one half the Nyquest frequency and with bandwidth ranging from 0+, which corresponds to a narrow pass band, to 1.0, which corresponds to an all-pass filter.

| | **0+** | **0.2** | **0.4** | **0.6** | **0.8** | **1.0** |
|---|---|---|---|---|---|---|
| **CAME** | 0.242 | 0.526 | 0.530 | 0.532 | 0.529 | 0.501 |
| **SOMT** | 0.261 | 0.560 | 0.562 | 0.563 | 0.559 | 0.486 |

Table 4.6 contains these results. The interesting thing to note about these results is that the mid-range frequencies turn out to be as or, in the case of SOMT, more informative than the high frequencies. This is likely an artefact of the lossy JPEG compression which discards the high frequency information. We were not able to acquire images that had not been compressed.

We next investigate which regions of the image are the most effective for frequency based analysis. To further localize the analysis, we partition the images into 108 non-overlapping 128x128 pixel blocks. A band-pass filter with a bandwidth of 0.6 is then applied to each block and the remaining energy is linearly regressed against the transmissometer readings. These results are displayed visually in figures 4.4 and 4.5 in which the blocks with higher $R^2$ values are made to appear more red. As expected, the sky regions do not contain high frequency information and therefore are not effective for estimating visibility. In CAME, the most informative regions contain the horizon. However, in SOMT, the most informative regions are those just below the horizon. These regions contain lots of detail and are distant enough to be affected by the intervening atmosphere (unlike the regions just below the horizon in CAME which also contain lots of detail but are too close). There is also a nice gradient present in the $R^2$ values as the blocks get

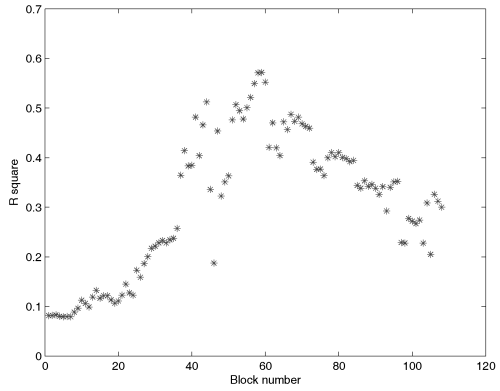



**Figure 4.3:** Plot of $R^2$ versus block number for SOMT for $128 \times 128$ blocks.

closer to the camera. This gradient is also evident in figure 4.3 which plots $R^2$ versus block number. The blocks are numbered one to 108 in raster scan order. Note how $R^2$ decreases with block number starting from approximately the middle of the image.

**Figure 4.4:** The regions in CAME most effective for frequency based analysis. Blocks with higher $R^2$ values are made to appear more red.

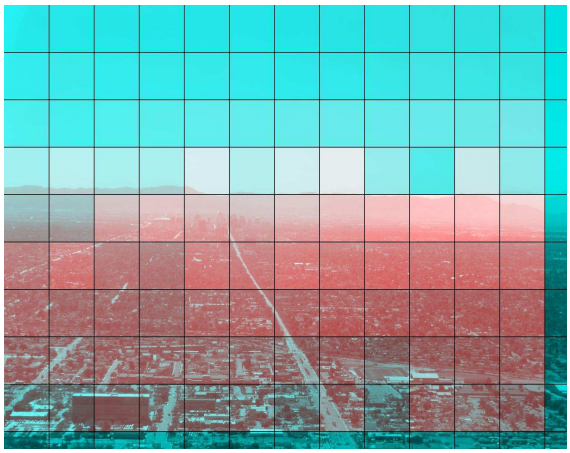


**Figure 4.5:** The regions in SOMT most effective for frequency based analysis. Blocks with higher $R^2$ values are made to appear more red.

# Chapter 5

# Semi-Supervised Regression with Temporal Image Sequences

*Work done in collaboration with Professor Miguel Á. Carreira-Perpiñán. Results published in The International Conference on Image Processing 2010* [43]

Motivated by the challenging problem and dataset described above, and under the assumption that the transmissometer values should vary slowly along a temporal sequence, we consider a semi-supervised regression problem where we have temporal sequences in which not all the training data is labelled. In particular, we have images every 15 minutes but transmissometer readings only every hour. We investigate whether the unlabeled images can be incorporated into the training to learn a more accurate model.

## 5.1 Methodology

Assume we have a training set of $N$ labeled points $\{\mathbf{x}_n, \mathbf{y}_n\}_{n=1}^N$ and $M$ additional unlabeled points $\{\mathbf{x}_m\}_{m=1}^M$. All $M+N$ inputs come as a collection of sequences of the form $(\mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3, \dots)$, each of which is only partially labeled with $\mathbf{y}$-values. We want to

estimate a regression mapping $\mathbf{f}$ that predicts the label $\mathbf{y}$ for an input $\mathbf{x}$. We consider a least-squares regression setting with a graph Laplacian regularization [44, 45]:

$$E(\mathbf{f}) = \sum_{n=1}^{N} \|\mathbf{y}_n - \mathbf{f}(\mathbf{x}_n)\|^2 + \gamma \|\mathbf{f}\|_K^2 + \gamma_I \|\mathbf{f}\|_G^2 \tag{5.1}$$

where the $\|\mathbf{f}\|_K^2$ term refers to an RKHS norm (which encourages smoothness irrespectively of the training data distribution), and the $\|\mathbf{f}\|_G^2$ term refers to the graph Laplacian (which encourages smoothness of $\mathbf{f}$ with respect to distribution of both labeled and unlabeled training points). The graph Laplacian is $\mathbf{L} = \mathbf{D} - \mathbf{W}$, where $\mathbf{W}$ is a given affinity matrix of $(N+M)\times(N+M)$ (such as the adjacency matrix or a Gaussian affinity matrix), and $\mathbf{D} = \mathrm{diag}\,(\mathbf{W1})$ is the degree matrix. Ordinarily one might learn a neighborhood graph in an unsupervised way, such as the $k$-nearest-neighbor graph for a suitable value of $k$. Here we propose to construct the graph as the collection of input sequences in $\mathbf{x}$-space. Thus, if all $M+N$ points are indexed in order by sequence, the corresponding adjacency matrix will contain ones in the sub- and super-diagonals (except where a sequence ends or starts), and zeroes elsewhere. The sequential regularization term is then quadratic on the label values $\mathbf{f}(\mathbf{x}_n)$:

$$\gamma_I \|\mathbf{f}\|_{G_t}^2 = \gamma_I \mathbf{f}^T \mathbf{L}_t \mathbf{f} = \sum_{\text{sequences}\, s} \sum_{n=2}^{N_s} w_{n,n-1} \|\mathbf{f}(\mathbf{x}_n) - \mathbf{f}(\mathbf{x}_{n-1})\|^2 \tag{5.2}$$

which is to be compared with the usual graph Laplacian term:

$$\gamma_I \|\mathbf{f}\|_G^2 = \gamma_I \mathbf{f}^T \mathbf{L} \mathbf{f} = \sum_{n \sim m} w_{nm} \|\mathbf{f}(\mathbf{x}_n) - \mathbf{f}(\mathbf{x}_m)\|^2 \tag{5.3}$$

Note this is not an additional graph prior term to the $k$-nearest-neighbor Laplacian, but that it replaces it. Although it is possible to use higher-order temporal priors (i.e., more

neighbors within each sequence), in this work we focus on linking consecutive points only. Our regularization term has the obvious interpretation of the squared gradient of $\mathbf{f}$ integrated along each sequence; thus, it cares about directional derivatives along paths in $\mathbb{R}^L$ rather than about the full gradient (which may be poorly constructed from the available samples in the problems we consider here).

If the points along a sequence were also nearest neighbors, then the $k$-nearest-neighbor graph with a low value of $k$ (perhaps even $k = 1$) would coincide or be very similar to our sequential graphs. However, in the problem of visibility estimation and others, the sequential structure is obscured by other neighboring points that greatly differ in label value. This is clear in our later experiments, where the best value of $k$ with the Laplacian prior is not very small. That is, for this type of problem, no value of $k$ (or $\epsilon$ if we use an $\epsilon$-ball graph) will yield a graph similar to the sequential graph. (In our experiments we have focused on 1D output values, but the method carries over to multidimensional outputs in a straightforward way.)

The solution of this regularized least squares problem carries over analogously to the original one but replaces the graph Laplacian matrix with $\mathbf{L}_t$, the one constructed from our sequences. The solution is unique and is given by a basis function expansion (depending on the RKHS) at each of the labeled and unlabeled points:

$$\mathbf{f}(\mathbf{x}) = \sum_{n=1}^{N+M} \boldsymbol{\alpha}_n K(\mathbf{x}_n, \mathbf{x}) \tag{5.4}$$

and the weights $\boldsymbol{\alpha}_n$ of this expansion are given by the solution of a linear system of size $(M + N) \times (M + N)$. In this part we use Gaussian kernels of width $\sigma$. As described, we

fix the loss function to the squared regression error, which results in a simple algorithm. Other formulations are possible with our regularization, such as a hinge loss.

## 5.2 Experimental Results

We performed three experiments using both synthetic and real world image datasets, and finally the proposed algorithm is tested on the visibility dataset. Each experiment compares the results of the proposed method, temporal Laplacian regularized least squares regression (TLapRLSR), with standard Laplacian regularized least squares regression (LapRLSR). Radial basis function (RBF) kernels are used in all cases.

### 5.2.1 Synthetic data: diverging spirals in three dimensions

The diverging spiral problem is shown in figure 5.1. The data points are specified to form two spirals that are interleaved in the two dimensional input space $(x, y)$ but diverge in the dimension $z$, which is the label to be predicted. The $z$ values of one of the spirals increase away from zero while the $z$ values of the other spiral decrease away from zero. We give 25 points for each spiral, for a total of 50 points. The training set consists of 10 labeled points for which the $z$ values are specified and 40 unlabeled points for which only the $(x, y)$ values are specified. The labeled points are approximately equally spaced along the spirals. Figure 5.1 shows the distribution of the labeled points.

For TLapRLSR, two "temporal" sequences are created, one for each spiral, in which adjacent points along the spiral are connected. This results in a 50 by 50 binary ad-

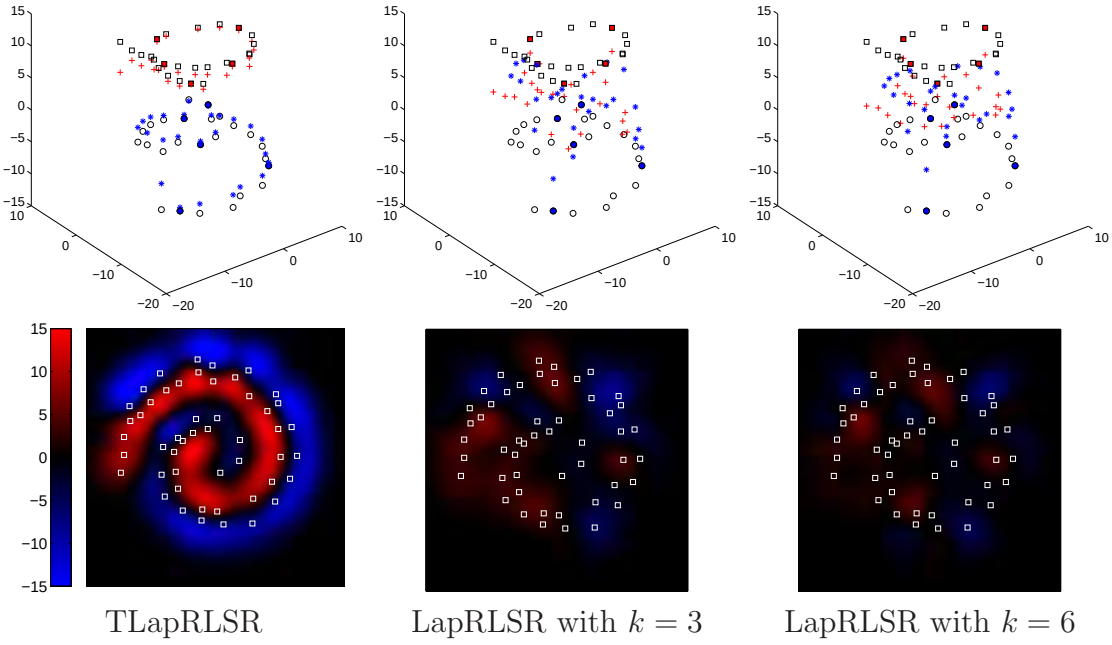



**Figure 5.1:** The diverging spiral problem. *Top*: 3D view of the data and method predictions. The boxes □ indicate the up sequence, whose $z$ values increase away from zero, and the circles ○ the down sequence, whose $z$ values decrease away from zero. The solid boxes/circles ■/● indicate the labeled training data points and the hollow ones □/○ the unlabeled training data. The pluses + and asterisks ∗ indicate the predicted $z$ values for the up and down sequences, respectively. *Bottom*: contour plot of the function learned by each method.

jacency matrix with 48 nonzero values. The binary adjacency matrix for LapRLSR is constructed using the $k = 6$ nearest neighbors with respect to Euclidean distance in the two dimensional input space (other $k$ gave similarly poor results). The methods are compared based on how well they predict the values of the unlabeled points (whose $z$ values are known but not provided to the training algorithms). The results in figure 5.1 show that prior information on the structure of the sequences allows TLapRLSR to more accurately predict the $z$ values of the datapoints. The reason is that the label does change smoothly in input space along each spiral, but sharply changes radially as one crosses different spiral branches. With sparse labeled data in the $(x, y)$ space, looking for nearest neighbors in Euclidean distance is as likely to pick points from either spiral. Note how the LapRLSR predictions form clusters centered near labeled points where the label changes smoothly and completely misses the spiral structure. However, TLapRLSR does learn the interleaved spirals; note its spirals are darker in the center, where the label is around zero, and brighter away from it, where the label grows in magnitude; they are also darker in the boundary between the spirals, where the label crosses from negative to positive.

### 5.2.2 Image data: rotated digits

This experiment investigates the challenging task of estimating the orientations of handwritten digits that are very similar except for a fixed rotation. For example, consider the digits in figure 5.2. It is difficult even for a human observer to tell whether the digits are a six at $x$ degrees or a nine at $x + 180$ degrees (the sixes have a curvy stroke, while the

**Figure 5.2:** What are the angles of these digits? Are they sixes at $x$ degrees or nines at $x+180$ degrees?

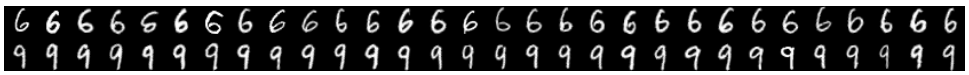


**Figure 5.3:** The 30 sixes and 30 nines from the MNIST dataset used to derive the rotated sequences.

nines have a straight one). Our dataset consists of rotated versions of 30 sixes and 30 nines from the MNIST database (which is a hand-written digits dataset) shown in figure 5.3. Each of the 30 nines are rotated counter-clockwise at one degree intervals from 0 to 180 degrees (labels are 0 to 180 degrees) and each of the 30 sixes are rotated counter-clockwise at one degree intervals from 180 to 360 degrees (labels are 180 to 360 degrees) for a total of 10 860 images. This results in a dataset in which *a six might appear very similar to a nine except for a 180 degree phase difference*. Therefore, Euclidean distances in image space would consider an upright 9 and an upside-down 6 as neighbors even though their labels differ drastically. Figure 5.4 demonstrates this for a particular pair of six and nine sequences.

The every-degree dataset is subsampled at every three and every five degrees to create two additional datasets for investigating the effect of the relation of the within-to-between sequence distances. Training, cross-validation and evaluation sets are constructed from the 60 sequences as follows. The elements of each sequence are assigned in an alternating fashion to training, cross-validation and evaluation sets resulting in a training set containing $61 \times 60 = 3\,660$ images, and cross-validation and evaluation sets containing

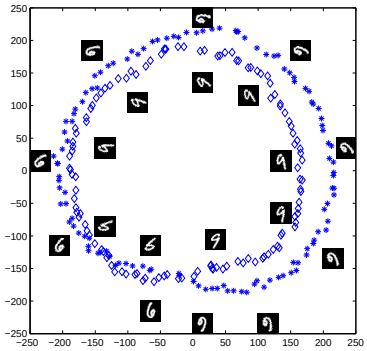


**Figure 5.4:** Images corresponding to a rotated six (asterisks $*$) and a rotated nine (diamonds $\blacklozenge$) sequence projected onto the first two PCA components. Both sequences start at the bottom and go counter-clockwise. The sequences are actually (parallel) spirals in the full three dimensional PCA space.

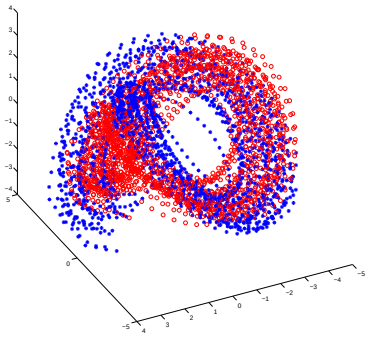


**Figure 5.5:** The rotated sequences of sixes (red) and nines (blue) projected onto the first three PCA components.

$60 \times 60 = 3\,600$ images each for the every-degree dataset. This is reduced to 1 220–1 200–1 200 and 732–700–700 for the every three and every five degree datasets, respectively. A percentage of the training set is labeled (with the rotation angle) at approximately equal spacings with respect to angle. The 60 sequences of labeled and unlabeled images in the training set form the "temporal" sequences for TLapRLSR.

Comparisons are performed using both the native feature space, in which the distance between two images is the square-root of the sum of the squares of the pixel differences (Euclidean distance between the images treated as vectors), as well as in a reduced 3D space, in which the images are projected onto the first three principal components computed over the training set and the difference between two images is the square-root of the sum of squares of the projection differences (Euclidean distance between the projected values). We refer to this reduced spaced as the principal component analysis (PCA) space. Figure 5.4 shows the distribution of the images corresponding to a particular pair of six and nine sequences in the two dimensional space formed by the first two principal components. Note again how samples from the different sequences can be closer to each other in the feature space than they are to samples from the same sequence even though there is a 180 degree phase shift. This scenario becomes even more likely with the full 60 sequences. Figure 5.5 plots all 60 sequences in the 3D PCA space. Note how the sixes' and nines' sequences interleave in a complex way, making it extremely hard to estimate the rotation angle. Although the angle along a sequence does vary smoothly, slight deviations outside the sequence can produce large angle deviations, and the smoothness assumptions built into a usual graph Laplacian may not hold well here. Figure 5.6 shows

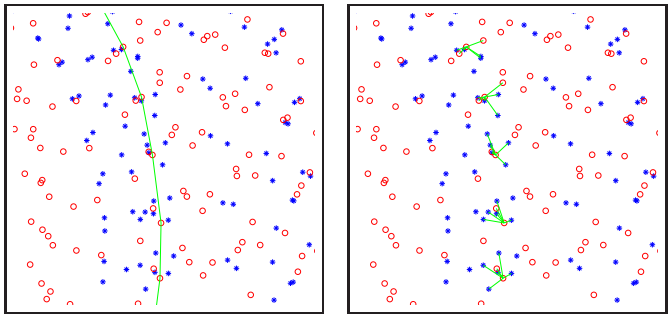


**Figure 5.6:** Examples of graphs constructed for the rotated digits problem by the proposed approach using the sequence information (left) and the standard approach with 6 nearest neighbors (right). Circles ○: sixes, asterisks ∗: nines, edges in green.

a 2D view of how the different approaches construct the graphs: the $k$-nearest-neighbor graph links both sixes and nines (thus encouraging them to have the same label, even though it differs by 180 degrees), while the temporal graph does not make that mistake.

We determined optimal parameter values using cross-validation. For both the temporal and $k$-nn approaches we set $\gamma_A = 10^{-3}$ (this value was shown to give reasonable results for both approaches). For the $k$-nn approach we initially set $k = 6$ for constructing the adjacency matrix following the approach of [44]. Optimal values for the remaining two parameters, $\gamma_I$ and $\sigma$, the width of the RBF kernel, are determined using the training and cross-validation sets. Figure 5.7 shows the prediction error for ranges of values of these parameters for the temporal and $k$-nn approaches for the native space for the rotated every one degree dataset. The plots for the PCA space are similar. It is clear that the temporal information reduces the prediction error, especially towards the right of the plots, where larger values of $\gamma_I$ mean the graph Laplacian is more heavily weighed.

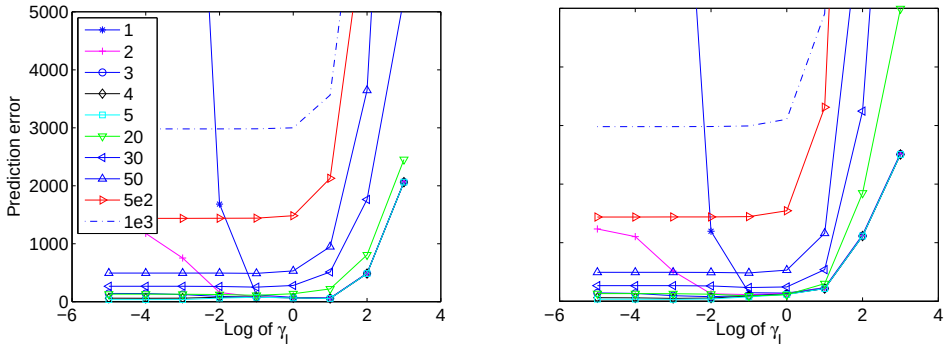



**Figure 5.7:** Prediction errors in regions bracketing the optimal free parameter values for TLapRLSR (left) and LapRLSR (right) for the every-one-degree rotated digit problem in the native space.

Figure 5.8 compares the predicted versus true angles for the test images for the every-one-degree rotated digits. It is very remarkable that TLapRLSR yields almost perfect prediction except in two small areas where it makes some large errors, while LapRLSR makes quite large errors almost everywhere (compare with the spirals' result). This results in TLapRLSR having always a smaller mean absolute error, even though sometimes LapRLSR does achieve the better mean squared error. Table 5.1 lists the prediction errors achieved by the optimal parameter settings and for different numbers of neighbors $k$ for the LapRLSR approach. These results correspond to a training set in which 20% of the images are labeled. Figure 5.9 shows the dependence of the two approaches on the percentage of labeled data for the every three degree rotated digits problem. Also shown is the prediction error for the fully supervised case, in which only the labeled training data is used (this is equivalent to setting $\gamma_I = 0$, effectively dropping the graph Laplacian term from the objective function). Learning is performed ten times for each ratio value

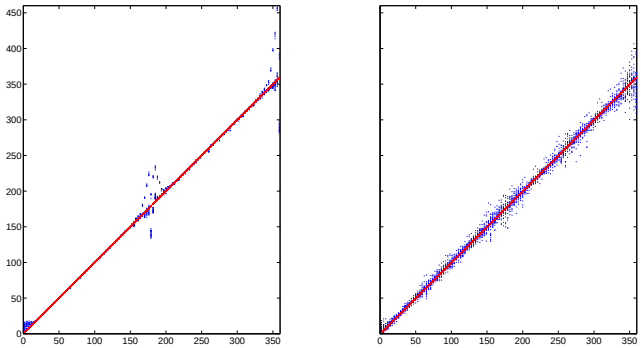



**Figure 5.8:** Predicted (Y–axis) versus true (X–axis) angle for the test images for TLapRLSR (left) and LapRLSR (right) for the every-one-degree rotated digits problem. The nines are plotted from 0 to 180 degrees and the sixes from 180 to 360 degrees. Perfect performance corresponds to the diagonal red line.

using randomly perturbed training sets. The error bars in the plot indicate the standard deviation of the prediction error. Note that our proposed TLapRLSR approach results in a significantly lower prediction error for a broad range of labeled data ratios. The margin of improvement often increases as the ratio of labeled data decreases, as shown for the PCA space here, which makes the proposed approach particularly attractive for the practically important problems were there is very little labeled data. The increase for TLapRLSR for high ratios for the PCA space indicates that the value $\gamma_I$ that was found to be optimal for a ratio of 0.2 starts to penalize the proposed approach as the ratio of labeled points increases. (Note the plot shows the mean squared error, which was the objective function of eq. (5.1), and TLapRLSR is not always the winner under that criterion; however, TLapRLSR does consistently beat LapRLSR and fully-supervised training with the mean absolute errors.)

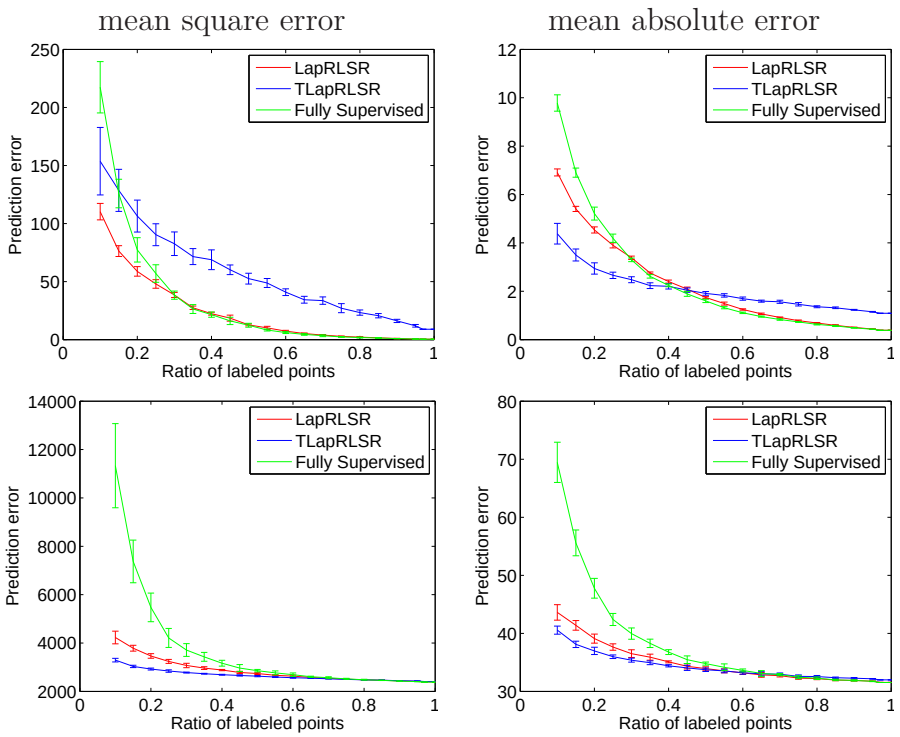



**Figure 5.9:** Dependence of the prediction error on the ratio of labeled points in the every-one-degree rotated digit problem, in the native (top row) and PCA (bottom row) spaces.

### 5.2.3 Image data: estimating time of day

This experiment investigates the problem of estimating the time of day (TOD) at which an image is captured. The dataset consists of 870 grayscale images of an outdoor scene acquired by a static camera at one minute intervals from 4:40am to 7:15pm. This setting for collecting data is very similar to the visibility dataset. Figure 5.10 shows images from 5:49 am, 5:20 pm, and 5:49 pm. The images are approximately evenly divided into training, cross-validation, and evaluation datasets. Approximately 20% of the training images are labeled (with the time of day) at equal spaced time intervals. The images in the training set form the single temporal sequence for TLapRLSR. Temporal information is critical for incorporating unlabeled images into the learning process for this problem since (1) parts of the scene might change rapidly over short time intervals due to clouds that obscure the sun or enter the scene, other objects, etc., and (2) images spaced far apart in time, such as from the morning and afternoon, can look similar. Figure 5.13 shows the affinity matrix for the $k$-nearest-neighbor graph and how it can miss correct edges or include wrong edges. Consecutive points with almost identical time labels often fail to be $k$-nn graph neighbors (missing near-diagonal entries); this is likely caused by short-term changes in the scenery or illumination. Conversely, points with very different, even extreme times are often $k$-nn graph neighbors (existing far-off-diagonal entries); this is particularly noticeable in the existing edges near the $(1, 250)$ elements, where the overall illumination is similar (dawn and dusk).

5:49 am 5:20 pm 5:49 pm

**Figure 5.10:** Three sample images from the time of day problem.

Again, comparisons are performed using both the native and reduced image spaces. The images are resized to $64 \times 64$ pixels. We set $\gamma_A = 10^{-3}$ and $k = 6$, and determine optimal values for the remaining two parameters, $\gamma_I$ and $\sigma$, through cross-validation. Figure 5.11 shows the prediction error for various values of these parameters for the temporal and standard approaches for the native space. The plots for the PCA space (not shown) are similar. Table 6.1 lists the minimum prediction errors achieved by the optimal parameter settings. TLapRLSR is again shown to outperform LapRLSR, both in mean squared error and mean absolute error, by a large margin (error two or three times smaller).

These results correspond to a training set in which 20% of the images are labeled. Figure 5.12 shows the dependence of the two approaches on the percentage of labeled data. Also shown is the prediction error for the fully supervised case, in which only the labeled training data is used. We can see that LapRLSR barely improves over the fully supervised setting, while TLapRLSR considerably improves over both over a wide range of the ratio of labeled points.

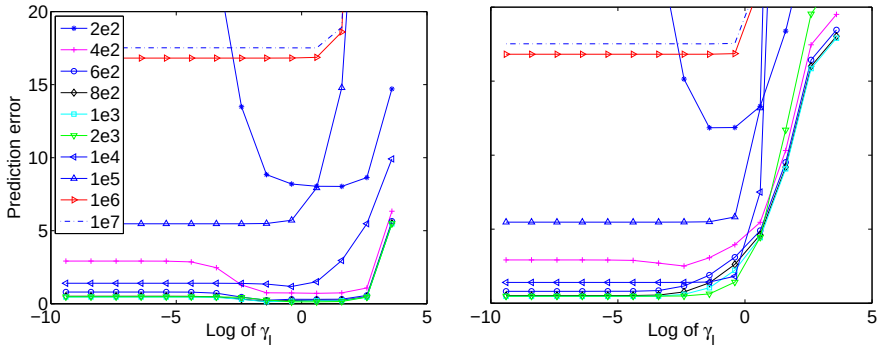



**Figure 5.11:** Prediction errors in regions bracketing the optimal free parameter values for TLapRLSR (left) and LapRLSR (right) for the time-of-day problem in the native space.

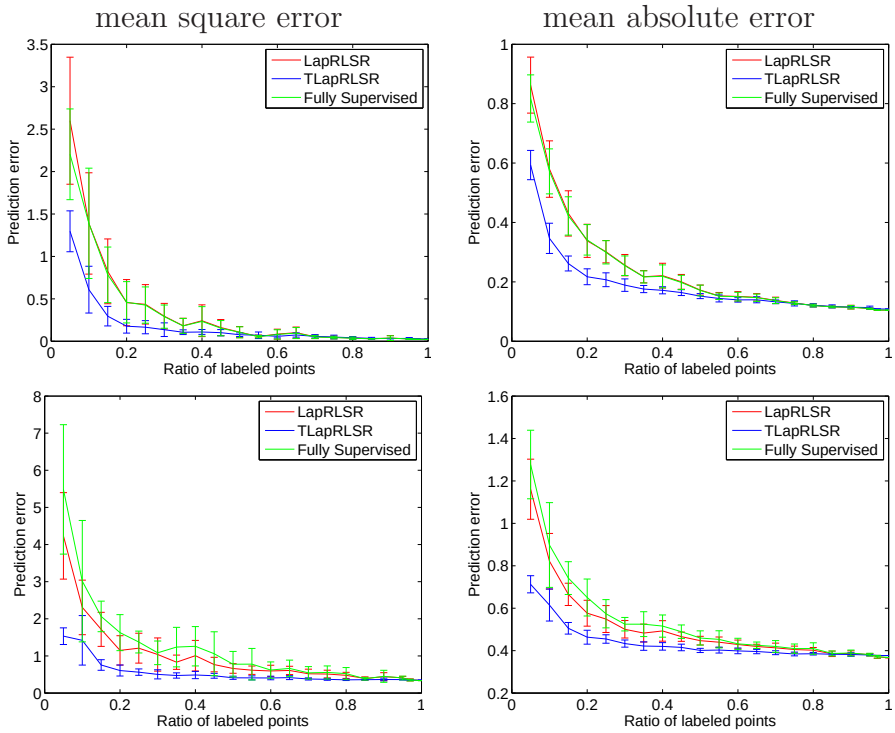



**Figure 5.12:** Dependence of the prediction error on the ratio of labeled points in the time-of-day problem, in the native (top row) and PCA (bottom row) spaces.

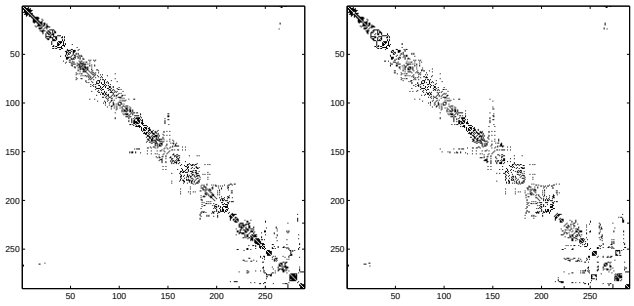



**Figure 5.13:** Time-of-day affinity matrix for the $k$-nearest-neighbor graph in the native (left) and PCA (right) spaces. The data points correspond to a single day and are ordered by increasing time (sunrise to sunset).

### 5.2.4 Image data: estimating visibility

We select 457 images from the visibility dataset (some images are missing) of which 123 are labeled. Every other labeled image is set aside as the test dataset. The remaining labeled and unlabeled images form the 14 temporal sequences for TLapRLSR.

Comparisons are performed using both the native and reduced image spaces. The images are resized to 128x128 pixels.

We set $\gamma_A = 10^{-3}$ and $k = 6$, and determine optimal values for the remaining two parameters, $\gamma_I$ and $\sigma$, through cross validation on the test set. Figure 5.14 shows the prediction error for various values of these parameters for the temporal and standard approaches for the native space. The plots for the PCA space are similar. Table 6.1 lists the minimum prediction errors achieved by the optimal parameter settings. The proposed approach again outperforms the standard approach although this time by less of a margin.

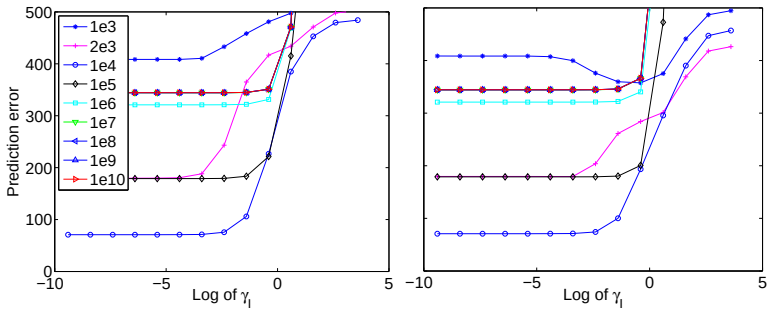



**Figure 5.14:** Prediction errors in regions bracketing the optimal free parameter values for TLapRLSR (left) and LapRLSR (right) for the visibility problem in the native space.

**Table 5.1:** Prediction errors achieved for rotated digits dataset with optimal parameter settings: mean square and, in parentheses, mean absolute error. Best results in boldface.

| | Rotated Digits | | | | | |
|---|---|---|---|---|---|---|
| | Every 1 Degree | | Every 3 Degrees | | Every 5 Degrees | |
| | Native | PCA $10^3$ ($10^0$) | Native | PCA $10^3$ ($10^0$) | Native | PCA $10^3$ ($10^0$) |
| TLap | 92.7 (**2.59**) | **2.73** (**35.6**) | 301 (**10.3**) | **3.67** (**47.9**) | 529 (**15.3**) | **3.88** (**45.9**) |
| Lap $k$=1 | 189 (8.47) | 3.33 (40.1) | 331 (13.0) | 4.36 (50.9) | 2140 (33.3) | 4.65 (53.5) |
| Lap $k$=2 | 32.9 (2.88) | 3.18 (38.5) | **240** (10.9) | 4.15 (49.4) | 1082 (23.5) | 4.59 (53.2) |
| Lap $k$=3 | 49.2 (4.26) | 3.13 (38.0) | 258 (11.3) | 4.08 (49.1) | 716 (10.5) | 4.54 (53.1) |
| Lap $k$=4 | **27.3** (2.87) | 3.09 (37.5) | 279 (11.6) | 4.06 (49.0) | 613 (18.6) | 4.50 (53.0) |
| Lap $k$=5 | 42.7 (3.92) | 3.08 (37.4) | 282 (11.7) | 4.04 (49.0) | 541 (17.7) | 4.50 (53.0) |
| Lap $k$=6 | 41.7 (3.82) | 3.06 (37.2) | 295 (12.1) | 4.03 (49.0) | 494 (16.9) | 4.50 (53.0) |
| Lap $k$=7 | 52.0 (4.42) | 3.04 (37.1) | 294 (12.1) | 4.02 (49.0) | **449** (15.9) | 4.48 (53.0) |

**Table 5.2:** Prediction mean square errors achieved for TOD and visibility datasets, with optimal parameter settings. Best results in boldface.

| | Time of Day | | Visibility | |
|---|---|---|---|---|
| | Native | PCA | Native | PCA |
| TLap | **0.111** | **0.568** | **116.7** | **159.2** |
| Lap $k$=1 | 0.346 | 1.07 | 120.8 | 161.6 |
| Lap $k$=2 | 0.330 | 1.02 | 119.0 | 160.0 |
| Lap $k$=3 | 0.329 | 1.03 | 119.5 | 159.8 |
| Lap $k$=4 | 0.332 | 0.967 | 120.4 | 159.9 |
| Lap $k$=5 | 0.345 | 0.930 | 119.6 | 159.9 |
| Lap $k$=6 | 0.348 | 0.930 | 118.3 | 160.1 |
| Lap $k$=7 | 0.349 | 0.995 | 118.0 | 160.4 |

# Chapter 6

# Estimating How Scenic An Image Is

*Results submitted to ACM Multi-Media Workshop on Social Media, 2011*

ScenicOrNot[1] is a project in which the public rate the "scenicness" of images from the *Geograph UK project*. The goal of the *Geograph UK project*[2] is to collect geographically representative photographs and information for every square kilometer of Great Britain and Ireland, and it currently contains 1,817,702 images. So far, the ScenicOrNot dataset contains 217,000 images with a minimum of 3 ratings each. The range of the rating is from 1 to 10. The manually labeled ScenicOrNot images are being used to generate a scenicness map for real estate purposes in the UK. Being able to automatically estimate the scenicness of other images would increase the scope and accuracy of such a map as well as enable other applications.

[1] `http://scenic.mysociety.org/`
[2] `http://www.geograph.org.uk/`

**Figure 6.1:** Examples from ScenicOrNot dataset. Left is one image example with rating 1 or 'not scenic', and right is one image example with rating 10 or 'very scenic'.

## 6.1 Dataset

We use the ScenicOrNot API to download a large number of images and their ratings and geotags. Among the downloaded images, only those with more than six ratings and small variation in ratings are considered. 105K images remain after this filtering. Then, the range of the ratings is mapped from [1...10] to [1...5], which means every 2 rating levels are pooled into one rating bin. Our final dataset contains [20851, 40578, 30959, 11845, 1145] images for each rating level. Examples of images with high and low rating are shown in figure 6.1.

## 6.2 Baseline Experiment Results

We perform a simple experiment to see how well a classifier can estimate the scenicness of an image based on its visual content. We choose simple gist features from [21] extracted globally from each image. Each image is represented as a 60 dimensional feature vector. Since there are relatively few images with rating '5', an instance balancing strategy is

**Table 6.1:** Classification results for ScenicOrNot dataset(%).

| Whole dataset | First 10K | First 1k | Random 20k | | | |
|---|---|---|---|---|---|---|
| | | | No SMOTE | SMOTE(k=50) | SMOTE(k=20) | SMOTE(k=10) |
| 38.39 | 38.34 | 33.8 | 38.6 | 35.95 | 36.09 | 35.31 |

applied to the dataset. We use a method which is called synthetic minority over-sampling technique (SMOTE) [46]. For the classifier, we choose an SVM with an RBF kernel, and use cross-validation to evaluate the performance.

We divide the dataset into 10 subsets, and run the cross-validation 10 times, and report the average accuracy. Without applying SMOTE, we ran the experiment on the whole dataset, on the first 10K images, on the first 1K images, and on random set of 20K images. As presented in table 6.1, the result on randomly selected 20K images is very close to the result on the whole dataset. Consequently we apply SMOTE only on the randomly selected images, and set the parameter k (the number of neighbors in SMOTE) to 10, 20, 50, which results in an expanded training set. However, as shown in table 6.1, using SMOTE does not improve the performance.

From this baseline experiment, we can conclude that training the classifier on a randomly selected subset is as accurate as using the whole dataset, but at less cost. Also using SMOTE to expand the training set does not improve the performance, so it will not be utilized in further investigation into this problem.

## 6.3 Exploiting Location Information for Scene Interpretation

### 6.3.1 Methodology

We want to investigate the spatial relation between the images for scenicness estimation, and so we develop a joint visial-geographic location kernel. The use of composite kernels has been explored in [47] and [48]. However, these works focus on supervised classification using support vector machines. We instead focus on novel techniques for supervised and semi-supervised regression using composite kernels in a manifold regularization [45] framework.

Assume we have a training set of $N$ labeled geographic locations $\{(x_n, z_n, y_n)\}_{n=1}^{N}$, where $x_n \in \mathbb{R}^L$ is a visual observation (in our case, the visual features extracted from the images), $z_n \in \mathbb{R}^2$ is the two dimensional location, and $y_n \in \mathbb{R}$ is the label. Our goal is to learn a regression mapping $f$ that predicts the label $y$ for a geographic location based on either just the visual observation $x$ or both the observation and the spatial coordinates $z$.

We focus on regularized least squares regression (RLSR). We also consider semi-supervised regression in a graph Laplacian framework for the case where we have $M$ additional unlabeled locations $\{x_m, z_m\}_{m=1}^{M}$.

**Vis-Geo Kernel Regularized Least Squares Regression**

We incorporate both visual observation and location by extending the RLSR framework with a composite kernel. In particular, given a visual kernel $K_{vis}$ and a geographic kernel $K_{geo}$, we form a new kernel $\widetilde{K}$ as

$$\widetilde{K} = \beta K_{vis} + (1-\beta)K_{geo}; \quad (\beta \in [0,1]).$$

The motivation here is that we should consider both the visual features and the locations of the labeled training data when learning and applying the regressor.

We now have the objective function

$$\min_{\alpha,\beta}(Y - \widetilde{K}\alpha)^T(Y - \widetilde{K}\alpha) + \gamma\alpha^T\widetilde{K}\alpha. \tag{6.1}$$

where the first term corresponds to linear prediction error and the second corresponds to the regularization of the model.

For convenience, we denote $K = K_{geo} - K_{vis}$ and $\widetilde{K} = K_{geo} - \beta K$ and then get the solution

$$\beta^\star = \frac{\frac{\gamma}{2}\alpha^T K\alpha + \alpha^T K_{geo} K\alpha - Y^T K\alpha}{\alpha^T K^2\alpha}.$$

We employ line-search to find the optimal $\beta$ using a strong boundary to satisfy the constraint $\beta \in [0,1]$. The line-search range is set from $\beta_0$, which is the current $\beta$, to $\beta^\star$, with an offset.

Finally, to obtain the optimal solution to eq. 6.1, we alternately update $\alpha$ and $\beta$ until convergence.

**Vis-Geo Kernel Laplacian Regularized Least Squares Regression**

We now form a composite vis-geo graph Laplacian as well as a composite vis-geo kernel:

$$\widetilde{K} = \beta K_{vis} + (1-\beta)K_{geo}; \qquad \widetilde{L} = \beta L_{vis} + (1-\beta)L_{geo}; \qquad (\beta \in [0,1]) \tag{6.2}$$

The graph Laplacian $L_{geo}$ is based on a nearest neighbor graph in the two dimensional location space. The motivation is that unlabeled observations can be expected to have similar labels to visually similar and nearby labeled observations.

The cost function now becomes

$$\min_{\alpha,\beta}(Y-\widetilde{K}\alpha)^T(Y-\widetilde{K}\alpha) + \gamma\alpha^T\widetilde{K}\alpha + \frac{\gamma_I}{(N+M)^2}\alpha^T\widetilde{K}\widetilde{L}\widetilde{K}\alpha.$$

Again, we denote $\widetilde{K} = K_{geo} - \beta K$ and $\widetilde{L} = L_{geo} - \beta L$. Setting the derivative of cost function with respect to $\beta$ to zero, we get,

$$\begin{aligned}
&\frac{\partial\rho}{\partial\beta} = Y^TJK\alpha + \beta\alpha^TKJK\alpha - \alpha^TK_{geo}JK\alpha \\
&-\gamma_a\alpha^TK\alpha + \beta\gamma_I\alpha^T(KL_{geo}K + KLK_{geo} + K_{geo}LK) \\
&-\beta^2\gamma_I\alpha^TKLK\alpha \\
&= 0
\end{aligned}$$

which is quadratic and thus does not necessarily have a single solution. So letting

$$\begin{aligned}
A &= \gamma_I\alpha^TKLK\alpha \\
B &= -\alpha^TKJK\alpha \\
&\quad -\gamma_I\alpha^T(KL_{geo}K + KLK_{geo} + K_{geo} + LK) \\
C &= \alpha^TK_{geo}JK\alpha + \gamma_A\alpha^TK\alpha - Y^TJK\alpha
\end{aligned}$$

the quadratic equation becomes,

$$A\beta^2 + B\beta + C = 0 \tag{6.3}$$

$$\Delta = B^2 - 4AC.$$

Therefore, the solution is

$$\beta^\star = \begin{cases} 2\sqrt{AC} & \text{if } \Delta \leq 0 \\ \psi(\frac{-B \pm \sqrt{B^2-4AC}}{2A}) & \text{otherwise} \end{cases}$$

where $\psi$ is the selection function to get the $\beta$ closer to the range [0,1]. We again employ line-search to find the optimal $\beta$ using a strong boundary to satisfy the constraint $\beta \in [0, 1]$. The line-search range is set from $\beta_0$, which is the current $\beta$, to $\beta^\star$, with an offset. We again alternately update $\alpha$ and $\beta$ until convergence.

### 6.3.2 Experiment Settings

We produce scenicness maps of the entire UK using the following approaches:

**Manual** Interpolating the labels of the 1K manually labeled training set. Note there is no model or learning performed in this case.

**Vis RLSR** Using RLSR to learn a regressor using the visual features and labels of the 1K training set and then applying this regressor to the visual features of the 3K test images. The labels of the 3K test images are then interpolated to form the map.

**Vis LapRLSR** Using graph Laplacian RLSR to learn a regressor using the visual features and labels of the 1K training set and the visual features of the 2K unlabeled set. The

regressor is then applied to the visual features of the 3K test images and the labels are interpolated.

**Vis-Geo RLSR** Using RLSR to learn a regressor using the visual features, locations, and labels of the 1K training set and then applying this regressor to the visual features and locations of the 3K test images. The labels of the 3K test images are then interpolated to form the map.

**Vis-Geo LapRLSR** Using graph Laplacian RLSR to learn a regressor using the visual features, locations, and labels of the 1K training set and the visual features and locations of the 2K unlabeled set. The regressor is then applied to the visual features and locations of 3K test images and the labels are interpolated.

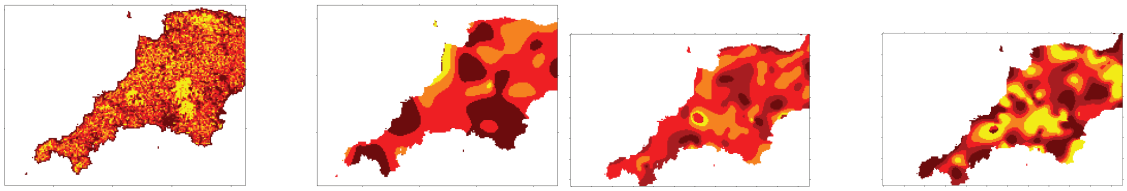
In order to compare the approaches, we observe that people's sense of scenicness correlates with openness and use a map of open countryside publicly available in the Countryside Information System of the UK Department for the Environment, Food, and Rural Affairs. This map which we term UKOC provides the percent "openness" for every square kilometer of the UK. We normalize it to the same scale as our scenicness labels and use it for qualitative evaluation through visual comparison and quantitative evaluation by computing the sum-square error with our interpolated labels.

### 6.3.3 Results

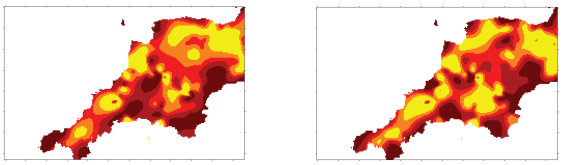
Table 6.2 shows the sum-square error between the UKOC values and the predicted scenicness values for the whole UK. Also shown are the errors for three subregions: the

**Table 6.2:** Sum-square error between the predicted labels and the UKOC values.

| | Manual | Vis RLSR | Vis LapRLSR | Vis-Geo RLSR | Vis-Geo LapRLSR |
|---|---|---|---|---|---|
| **UK** | 581.09 | 573.10 | 561.18 | 561.14 | **557.78** |
| **CORN** | 179.76 | 168.31 | 161.20 | 163.24 | **159.88** |
| **LON** | 125.81 | 120.71 | 114.38 | **107.13** | 111.71 |
| **LM** | 166.77 | 159.89 | 147.41 | 152.89 | **147.22** |



(a) Open Country (b) Manual (c) Vis RLSR (d) Vis LapRLSR

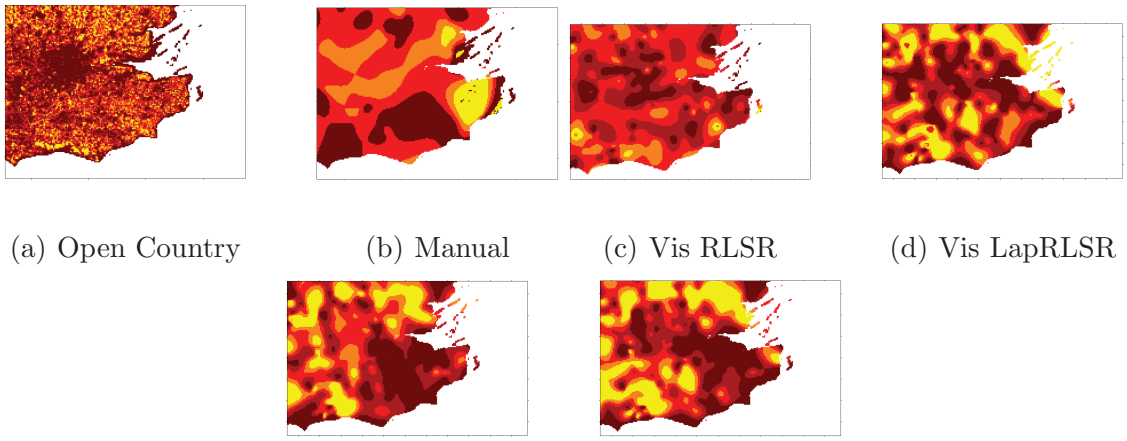


(e) Vis-Geo RLSR (f) Vis-Geo LapRLSR

**Figure 6.2:** (a) Ground truth and (b-f) predicted maps for the Cornwall area.

Cornwall area (CORN), the London area (LON), and the Liverpool and Manchester area (LM).

These results demonstrate that 1) we are able to learn a regressor that when applied to set of test images, results in a more accurate map than simply interpolating the manually labeled locations; 2) the combined vis-geo kernel performs better than a standard visual only kernel; and 3) using graph Laplacian regularization to perform semi-supervised learning using unlabeled images improves both the combined vis-geo and visual only kernel approaches in all cases but one.



(a) Open Country (b) Manual (c) Vis RLSR (d) Vis LapRLSR

(e) Vis-Geo RLSR (f) Vis-Geo LapRLSR

**Figure 6.3:** (a) Ground truth and (b-f) predicted maps for the London area.

The predicted maps for the two subregions are visually compared with the UKOC map in figures 6.2-6.3. While the predicted maps are noisy, we make the following observations. The CORN area has roughly three open country regions and one urban region as shown in figure 6.2(a). The map that results from interpolating the manually labeled image locations in figure 6.2(b) clearly fails to identify these regions. Vis RLSR produces the map in figure 6.2(c) which identifies one of the open country regions but otherwise is not accurate. Vis LapRLSR, Vis-Geo RLSR, and Vis-Geo LapRLSR identify all four regions but the Vis-Geo approaches generally provide better localization.

The LON area has one large urban region corresponding to greater London as shown in figure 6.3(a). This time, the interpolated manually labeled map does better but is still poor. Vis RLSR does identify the urban region but underestimates the values for the rest of the area. Vis LapRLSR, Vis-Geo RLSR, and Vis-Geo LapRLSR all identify the

urban region and outlying open country regions but the two graph Laplacian regularized approaches provide better localization especially south of London.

# Chapter 7

# Conclusion

## 7.1 Visibility Estimation

In chapter 4, the results demonstrate the potential for using general-purpose cameras to estimate atmospheric visibility. In particular, we show the correlation between features automatically extracted from the images and standard measures of visibility from transmissometers. $R^2$ values above 0.6 are achieved for two cameras from the Phoenix region. Our approach compares favorably with previous approaches although a direct comparison is difficult because we use transmissometer readings as the ground truth whereas most other approaches use human observations which are much coarser. Unlike other approaches, ours can be completely automated and is robust to camera movement.

We explore two approaches. In the first approach, image contrast is measured in the spatial domain using bands of pixels above and below the horizon. We show that it is better to use a narrow band of pixels especially if image registration is performed to correct for camera movement. If color images are available then the green channel is shown to be the best choice. In the second approach, the energy in different image

frequency bands is computed through filtering. We confirm that the atmosphere acts as a low-pass filter. We also demonstrate that distant regions containing lots of detail are the most effective for estimating visibility.

## 7.2 Temporal Semi-Supervised Regression

We have considered in chapter 5, a semi-supervised regression setting where the supervision information consists not only of the labels of a small proportion of the input patterns, but also of the neighborhood graph of the inputs as a collection of disconnected sequences. Our algorithm is a variation of Laplacian regularization for regression using this neighborhood graph. We have shown it to improve consistently over using as regularizer the $k$-nearest-neighbor graph of the inputs in cases where this neighborhood information is not a good predictor of the label—for example, when nearby inputs may have very different labels. This is particularly useful for problems with temporal structure and high-dimensional complex inputs, such as images, where it is very hard to tell which of the many things that change from image to image have an effect on the label. We expect our approach would also be useful for classification problems where the neighborhood information is not temporal but of other types.

## 7.3 Scenicness Estimation

In chapter 6, we described a method for using automatic scene understanding to create maps of the earth using large collections of georeferenced community contributed

photos. In particular, we demonstrated how a small set of manually labeled images can be used to train a regressor for predicting a more accurate map from additional images. We also describe a novel composite visual-geographic location kernel and demonstrated its improved performance over a purely visual kernel. Finally, we showed how unlabeled images can be incorporated in a graph Laplacian regularization framework using a composite visual-geographic location Laplacian.

# Bibliography

[1] Liu, J., Shah, M.: Scene modeling using co-clustering. In: Proc. of the 2007 IEEE International Conference of Computer Vision (ICCV'07). (2007)

[2] Oliva, A., Torralba, A.: Modeling the shape of the scene: A holistic representation of the spatial envelope. International Journal of Computer Vision **42** (2001) 145–175

[3] Caimi, F., Kocak, D., Justak, J.: Remote visibility measurement technique using object plane data from digital image sensors. In: Proceedings of the IEEE International Geoscience and Remote Sensing Symposium. (2004) 3288–3291

[4] Kim, K., Kim, Y.: Perceived visibility measurement using the HSI color different method. Journal of the Korean Physical Society **46** (2005) 1243–1250

[5] Baumer, D., Versick, S., Vogel, B.: Determination of the visibility using a digital panorama camera. Atmospheric Environment **42** (2008) 2593–2602

[6] Luo, C., Wen, C., Yuan, C., Liaw, J., Lo, C., Chiu, S.: Investigation of urban atmospheric visibility by high-frequency extraction: Model development and field test. Atmospheric Environment **39** (2005) 2545–2552

[7] Raina, D., Parks, N., Li, W., Gray, R., Dattner, S.: Innovative monitoring of visibility using digital imaging technology in an arid urban environment. In: Regional and Global Perspectives on Haze: Causes, Consequences and Controversies Visibility Specialty Conference. (2004)

[8] Molenar, J., Cismoski, D., Schreiner, F., Malm, W.: Analysis of digital images from Grand Canyon, Great Smoky Mountains, and Fort Collins, Colorado. In: Regional and Global Perspectives on Haze: Causes, Consequences and Controversies Visibility Specialty Conference. (2004)

[9] Nayar, S., Narasimhan, S.: Vision in bad weather. In: Proc. of the 1999 IEEE International Conference on Computer Vision (ICCV'99). (1999)

[10] Narasimhan, S., Nayar, S.: Chromatic framework for vision in bad weather. In: Proc. of the 2000 IEEE Computer Society Conf. Computer Vision and Pattern Recognition (CVPR'00). (2000)

[11] Schechner, Y., Narasimhan, S., Nayar, S.: Instant dehazing of images using polarization. In: Proc. of the 2001 IEEE Computer Society Conf. Computer Vision and Pattern Recognition (CVPR'01). (2001)

[12] Narasimhan, S., Nayar, S.: Removing weather effects from monochrome images. In: Proc. of the 2001 IEEE Computer Society Conf. Computer Vision and Pattern Recognition (CVPR'01). (2001)

[13] Jacobs, N., Roman, N., , Pless, R.: Consistent temporal variations in many outdoor scenes. In: Proc. of the 2007 IEEE Computer Society Conf. Computer Vision and Pattern Recognition (CVPR'07). (2007)

[14] Szummer, M., Picard, R.: Indoor-outdoor image classification. In: IEEE Intl Workshop on Content-based Access of Image and Video Databases.(ICCV'98). (1998)

[15] Serrano, N., Savakis, A., Luo, J.: Improved scene classification using efficient low-level and semantic cues. Pattern Recognition **37** (2004) 1773–1784

[16] Serrano, N., Savakis, A., Luo, J.: A computationally efficient approach to indoor/outdoor scene classification. In: IEEE International Conference on Pattern Recognition. (2003)

[17] Meine, A., Hermes, T., Ioannidis, G., Fathi, R., Herzog, O.: Automatic shot boundary detection and classification of indoor and outdoor scenes. In: Information Technology: The 11th Text Retrieval Conference. (2003)

[18] Guerin-Dugue, A., Olivia, A.: Classification of scene photographs from local orientation features. Pattern Recognition Letters **21** (2000) 1135

[19] Traherne, M., Singh, S.: An inergrated approach to automatic indoor-outdoor scene classification in digital images. In: Proc. 5th International Conference on Intelligent Data Engineering and Automated Learning. (2004)

[20] Payne, A., Singh, S.: Indoor vs. outdoor scene classification in digital photographs. Pattern Recognition **38** (2005) 1533–1545

[21] Oliva, A., Torrralba, A.: Modeling the shape of the scene: A holistic representation of the spatial envelope. International Journal of Computer Vision (2001)

[22] Oliva, A., Torrralba, A.: Scene-centered description from spatial envelope properties. In: Proc. 2nd Workshop on Biologically Motivated Computer Vision. (2002)

[23] Li, F., Perona, P.: A bayesian hierachical model for learning natural scene categorization. In: Proc. of the 2005 IEEE Computer Society Conf. Computer Vision and Pattern Recognition (CVPR'05). (2005)

[24] Sudderth, E., Torralba, A., Freeman, W., Willsky, A.: Learning hierarchical models of scenes, objects, and parts. In: Proc. of the 2005 IEEE International Conference of Computer Vision (ICCV'05). (2005)

[25] Kivinen, J., Sudderth, E., Jordan., M.: Learning Multiscale Representations of Natural Scenes Using Dirichlet Process. In: Proc. of the 2007 IEEE International Conference of Computer Vision (ICCV'07). (2007)

[26] Quelhas, P., Monay, F., Odobez, J., Gatica-Perez, D., Tuytelaars, T., Gool, L.V.: Modeling scenes with local descriptors and latent aspects. In: Proc. of the 2005 IEEE International Conference of Computer Vision (ICCV'05). (2005)

[27] Bosch, A., Zisserman, A., Munoz, X.: Scene classification via pLSA. In: Proc. of the 2006 European Conference of Computer Vision (ECCV'06). (2006)

[28] Bosch, A., Zisserman, A., Munoz, X.: Scene Classification Using a Hybrid Gernative/Discriminative Approach. IEEE Transaction on Pattern Analysis and Machine Intelligence **30** (2008) 712–727

[29] Lazebnik, S., Schmid, C., Ponce, J.: Beyond bags of features: Spatial pyramid matching for recognition natural scene categories. In: Proc. of the 2006 IEEE Computer Society Conf. Computer Vision and Pattern Recognition (CVPR'06). (2006)

[30] van Gemert, J., Geusebroek, J., Veenman, C., Smeulders, A.: Kernel codebooks for scene categorization. In: Proc. of the 2008 European Conference of Computer Vision (ECCV'08). (2008)

[31] Yang, J., Yu, K., Gong, Y., Huang, T.: Linear spatial pyramid matching using sparse coding for image classification. In: Proc. of the 2009 IEEE Computer Society Conf. Computer Vision and Pattern Recognition (CVPR'09). (2009)

[32] Wu, J., Rehg., J.: Beyond the Euclidean distance: Creating effective visual codebooks using the histogram intersection kernel. In: Proc. of the 2009 IEEE International Conference of Computer Vision (ICCV'09). (2009)

[33] Gao, S., Tsang, I., Chia, L., Zhao, P.: Local Features Are Not Lonely-Laplacian Sparse Coding for Image Classification. In: Proc. of the 2010 IEEE Computer Society Conf. Computer Vision and Pattern Recognition (CVPR'10). (2010)

[34] Yanai, K., Yaegashi, K., Qiu, B.: Detecting cultural differences using consumer-generated geotagged photos. In: Proceedings of the International Workshop on Location and the Web. (2009)

[35] Crandall, D., Backstrom, L., Huttenlocher, D., Kleinberg, J.: Mapping the world's photos. In: WWW. (2009)

[36] Jacobs, N., Satkin, S., Roman, N., Speyer, R., Pless, R.: Geolocating static cameras. In: IEEE 11th International Conference on Computer Vision. (2007)

[37] Cristani, M., Perina, A., Castellani, U., Murino, V.: Geo-located image analysis using latent representations. In: IEEE Conference on Computer Vision and Pattern Recognition . (2008)

[38] Leung, D., Newsam, S.: Proximate sensing: Inferring What-Is-Where from georeferenced photo collections. In: IEEE Conference on Computer Vision and Pattern Recognition . (2010)

[39] Xie, L., Chiu, A., Newsam, S.: Estimating atmospheric visibility using general-purpose cameras. In: International Symposium on Visual Computing (ISVC' 2008). (2008)

[40] Seinfeld, J., Pandis, S.: Atmospheric Chemistry and Physics: From Air Pollution to Climate Change. Wiley (2006)

[41] Fedorov, D., Fonseca, L., Kenney, C., Manjunath, B.: Automatic registration and mosaicking system for remotely sensed imagery. In: SPIE 9th International Symposium on Remote Sensing. (2002)

[42] Krishnakumar, V., Venkatakrishnan, P.: Determination of the atmospheric point spread function by a parameter search. Astronomy & Astrophysics Supplement Series **126** (1997) 177–181

[43] Xie, L., Carreira-Perpinan, M.A., Newsam, S.: Semi-supervised regression with temporal image sequences. In: Proceedings of the IEEE International Conference on Image Processing. (2010)

[44] Belkin, M., Niyogi, P., Sindhwani, V.: Manifold regularization: A geometric framework for learning from labeled and unlabeled examples. Journal of Machine Learning Research **7** (2006) 2399–2434

[45] Belkin, M., Niyogi, P., Sindhwani, V.: On manifold regularization. In: Proc. of the Tenth Int. Workshop on Artificial Intelligence and Statistics (AISTATS 2005). (2005)

[46] Chawla, N., Bowyer, K., Hall, L., Kegelmeyer, W.: SMOTE: Synthetic minority over-sampling techinque. Journal of Artificial Intelligence Research **16** (2002) 321–357

[47] Sonnenburg, S., Ratsch, G., Schafer, C.: A general and efficient multiple kernel learning algorithm. In: Advances in Neural Information Processing Systems. (2005)

[48] Bach, F.R., Lanchriet, G.R.G., Jordan, M.I.: Multiple kernel learning, Conic Duality and the SMO algorithm. In: Proc. of the 21st Int. Conf. Machine Learning (ICML'04). (2004)