WEAKLY SUPERVISED LEARNING ON IMAGE MANIFOLDS

by

Hui Wu

A dissertation submitted to the faculty of
The University of North Carolina at Charlotte
in partial fulfillment of the requirements
for the degree of Doctor of Philosophy in
Computing and Information Systems

Charlotte

2015

Approved by:

Dr. Richard M. Souvenir

Dr. Min C. Shin

Dr. Andrew R. Willis

Dr. Shaoting Zhang

Dr. Wlodek Zadrozny

ABSTRACT

HUI WU. Weakly supervised learning on image manifolds. (Under the direction of DR. RICHARD M. SOUVENIR)

Recent work in image manifold learning has shown the prevalence of unsupervised methods that provide compact representation and perceptually meaningful organization of images in certain types of natural image sets. However, in situations where a discriminant factor needs to be discovered from an image set in which multiple latent variation factors exist, unsupervised methods are often limited. Whereas, supervised manifold learning approaches can be robust against irrelevant factors by leveraging image labels which impose additional constraints on the relationships between images. Nonetheless, ground truth labels are usually too costly to obtain and sometimes not entirely available. In this dissertation, we are interested in learning on image manifolds with weak supervision. The weakly supervised learning methods that we present are capable of mitigating the manual labeling effort required by supervised methods. In particular, we consider three variants of weakly supervised learning on image manifolds: (1) image labels not explaining all latent factors of image variation, (2) image labels which are heavily corrupted, and (3) image labels being partly available. We propose an algorithmic solution for each problem and evaluate the performance of the proposed algorithms quantitatively and qualitatively on a wide range of data sets.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1: BACKGROUND

Computer vision methods aim to infer properties of the real world from image data. Figure 1.1 shows two typical computer vision problems: recognizing the actions of soccer players in a sport video, and making age estimates from faces seen in images. To represent images, a myriad of image features have been proposed; some examples are: raw pixel intensities, histograms of oriented gradients, and bag of visual words. Usually most image features in computer vision are very high-dimensional with thousands of dimensions or even more. In the general case, it is impossible to learn a reliable predictive model in such high-dimensional spaces. When data dimensionality increases, the volume of the feature space increases exponentially, which makes the size of typical data sets insignificant. This phenomenon is known as the "curse of dimensionality".

(a) Action recognition in sport videos

(b) Age estimation from face images

Figure 1.1: Examples of computer vision applications.

However, most image sets have intrinsic structures that can be expressed by a few latent factors. For example, a set of depth images depicting the movement of a hand can be represented in terms of the displacement and rotation of each joint (Figure 1.2). An image set of nearly frontal human faces containing changes in gender, age, pose, expression, etc (Figure 1.3). An image capturing a human action changes its appearance mainly due to the type of action being performed, the person specific style, and camera viewpoint (Figure 1.4). Although the specific image feature being used for each application can be very high-dimensional, the underlying structure of a given image set is usually governed by only a few variables.

Figure 1.2: A set of depth images of a moving hand [76]. The corresponding articulated hand poses are denoted by connected colored points.

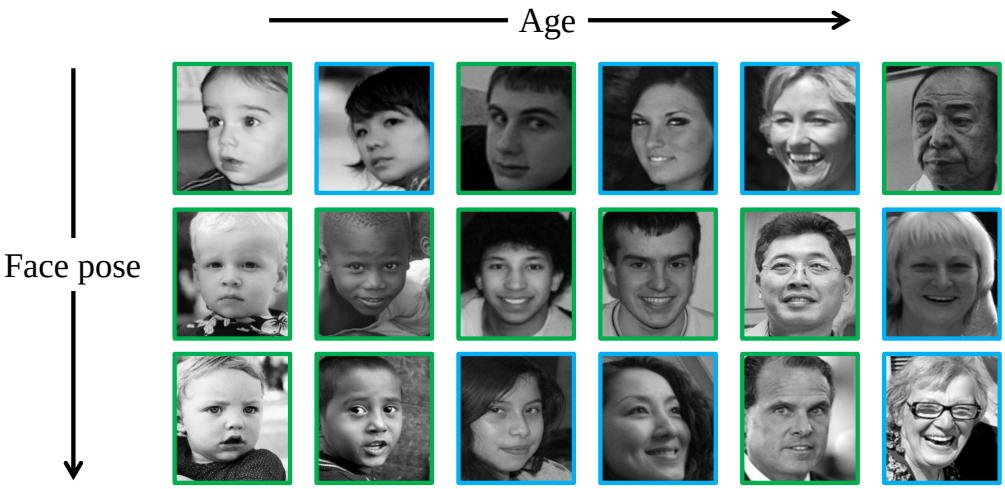


Figure 1.3: Example images from GeoFaces [48] data set. The image set contains factors such as pose, age, gender (indicated by colored boxes), etc.

Either implicitly or explicitly, most learning algorithms in computer vision exploit

this underlying structure to make learning and inference possible. In this dissertation, we focus on methods that explicitly utilize the low-dimensional structure of image sets. Discovering the low-dimensional representation automatically from images has many benefits, such as providing an insightful visualization of the image set, alleviating the "curse of dimensionality", and facilitating other tasks such as clustering and retrieval. When the latent factors are continuously changing (such as the face pose and age in Figure 1.3), manifold learning provides powerful computational approaches to discover the latent structure of image sets.

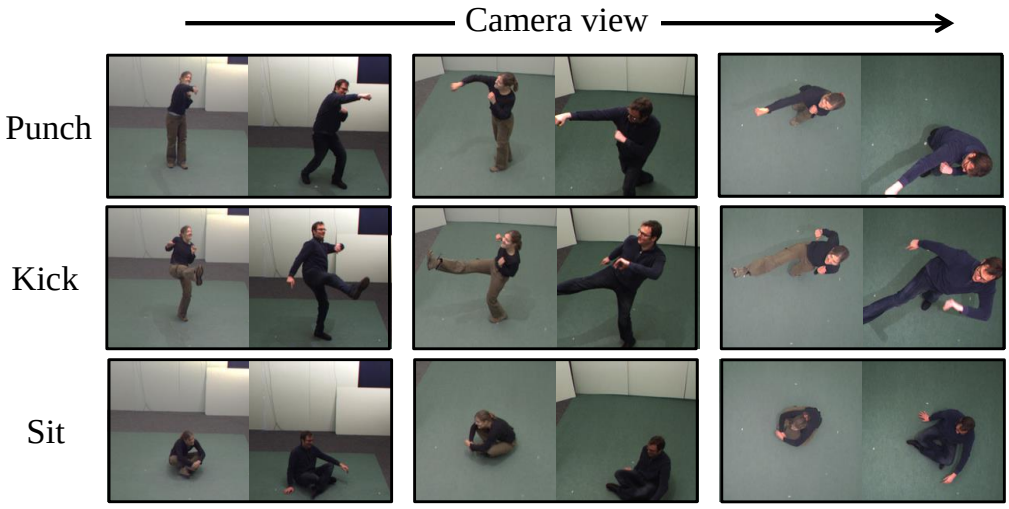


Figure 1.4: Example images from IXMAS [106], a commonly used data set for testing action recognition algorithms. Image variation is mainly due to changes in action class, actor identity, and camera view.

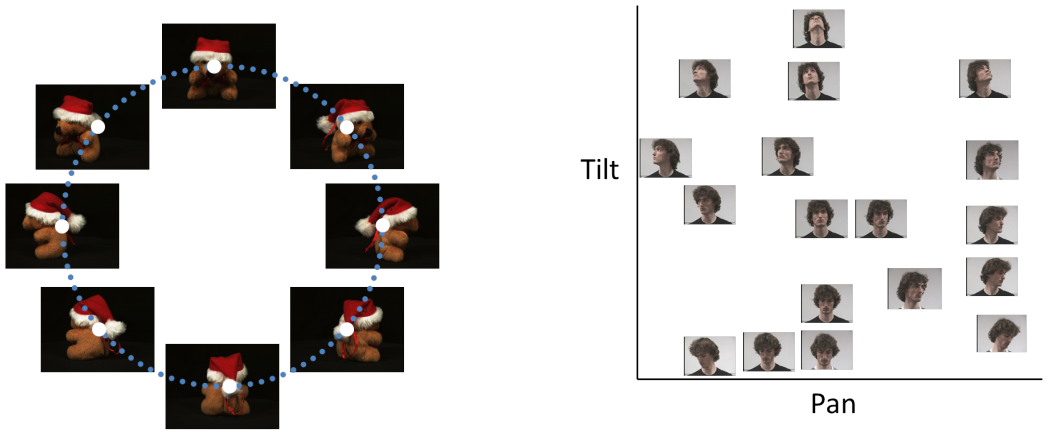
In many situations, not all of the low-dimensional factors are directly applicable for the intended task. For example, for age estimation, only image changes due to age are considered informative while other factors of change should be ignored. For the problem of human action recognition, the goal is to discriminate between different actions while discarding changes in human identity and camera viewpoint. Many

computer vision problems can be framed as supervised learning problems, where image labels relevant to the desired output are provided for each image. With the supervision provided by image labels, the factors of interest can be separated from irrelevant factors on the manifold.

Ideally, reliable image labels are available for each image in the data set. However, with the increasing number of large-scale image sets and the growing complexity of image labels, collecting full annotations for every image is very challenging. In this dissertation, we investigate problems in image manifold learning with weak supervision. Unlike supervised or unsupervised learning problems, weakly supervised learning can not be uniquely defined since both the type and the level of supervision may vary. In Chapter 1.2, we will introduce multiple variants of weakly supervised learning on image manifolds.

1.1 Image Manifold Learning

Figure 1.5 shows two example image sets, both of which can be organized according to perceptually meaningful factors. Recovering the low-dimensional representation automatically from high-dimensional data is referred to as dimensionality reduction. Classical techniques for dimensionality reduction assume that the variation caused by the underlying factors is mostly linear. So the low-dimensional parametrization can be approximated by the projection of high-dimensional points onto the learned linear subspace. Principal component analysis (PCA) [52] learns the linear subspace that maximally correlates with data variation and represents points as their coordinates in the learned lower-dimensional subspace.



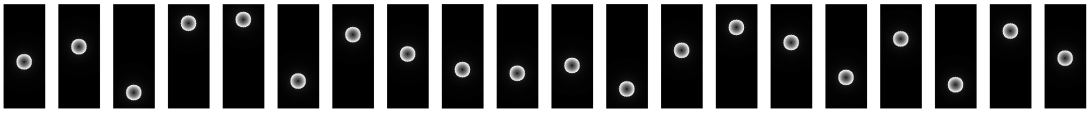
(a) A teddy bear observed from varying view angles (b) Images showing changing head poses

Figure 1.5: Many image sets in computer vision can be organized in a perceptually meaningful way in a low-dimensional space.

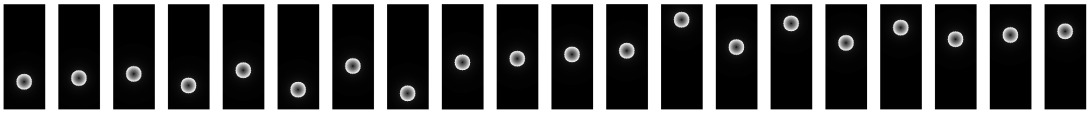
Although PCA is frequently used in many fields, the linearity assumption often fails to generalize to image variation, as demonstrated with a toy example in Figure 1.6. The coordinates of each image in the learned 1-D linear subspace are used to reorganize the images. As shown in Figure 1.6(b), PCA coordinates do not correspond with the latent factor, which, in this example, is the vertical position of the object. As can be seen in Figure 1.6(c), these images can not be approximated by a single parameter on a linear subspace, even though there is only a single underlying degree of freedom. This issue is even more evident in real-world data sets.

A large number of nonlinear dimensionality reduction techniques have been developed, which aim to address the limitations of classical linear methods. A large body of this work builds upon the notion that points in the high-dimensional space (or ambient space) lie on or near a nonlinear manifold with only a few degrees of freedom (intrinsic dimension). The low-dimensional representation of the data points

is computed by estimating the coordinate of each point on the underlying manifold. The problem is highly under-constrained, with unknown manifold structure and unknown intrinsic dimension. Various assumptions on the geometrical properties of the manifold have been proposed to constrain the problem [80, 90, 93, 101].

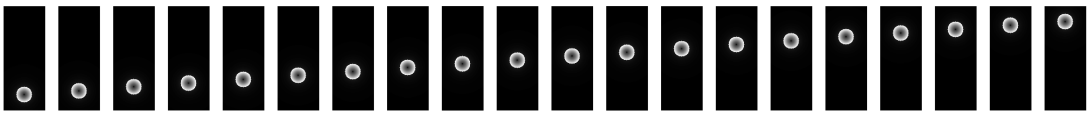


(a) Input image set



(b) Images organized by 1D PCA coordinates

(c) PCA approximations of each image in (b)



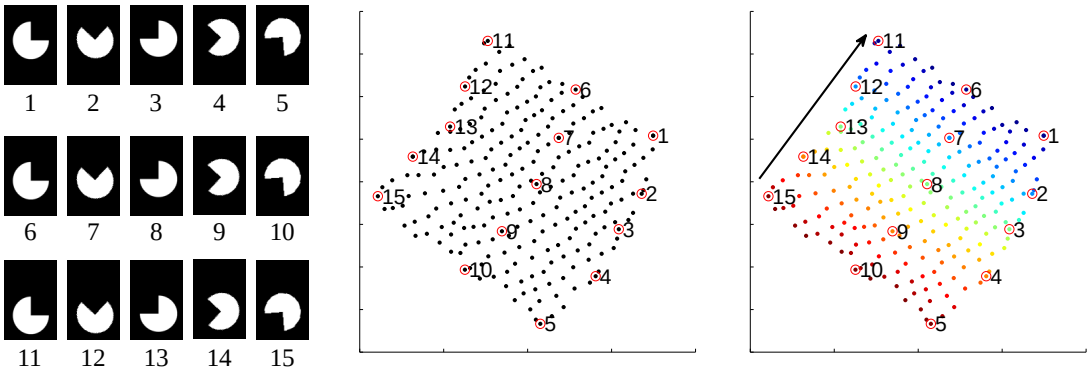
(d) Images organized by 1D Isomap [90] coordinates

Figure 1.6: Linear and nonlinear dimensionality reduction on a toy data set.

Compared with linear dimensionality reduction techniques, image manifolds often provide a more meaningful way to organize images. In Figure 1.6(d), the coordinates recovered by a manifold learning method accurately correspond with the latent factor of image change. Unlike linear dimensionality reduction, most manifold learning techniques do not parametrically model the structure of the data, but estimate the underlying manifold in a data-driven fashion. Consequently, manifold learning

methods often preserve relationships between images better than linear methods.

However, when there are multiple underlying factors of change, each dimension of the learned low-dimensional representation from manifold learning usually does not correspond with a semantically meaningful factor. For example, the image set shown in Figure 1.7 contains changes in rotation and translation. But in the recovered low-dimensional space, the directions of rotation change and translation change do not align with the axes (Figure 1.7(b)). In this case, when provided with image labels associated with one of the factors, changes caused by different factors can be well separated.



(a) Examples of the input (b) 2D embedding computed from Isomap

Figure 1.7: The input images change due to both rotation and translation. The recovered 2D representation from manifold learning, however, does not directly correspond with the two factors respectively. When image labels correlated with rotation are provided (denoted by the color of the points), it can supervise the learning process and help to separate the two factors.

In general, relevant information for the specific task, if available, can be incorporated to provide supervision and improve the performance of unsupervised methods. However, obtaining accurate labels for each image requires a lot of manual labeling effort. In some cases, with much less effort, partial or corrupted labels are available,

Table 1.1: Research problems investigated in this dissertation sorted by decreasing level of supervision.

Problem	**Label Amount**	**Label Corruption**	**Prior Knowledge on Variation Model**	**Chapter**
Weakly Sup. Manifold Factorization	Complete	Low	Known	3
Robust Manifold Regression	Complete	High	Unknown	4
Semi-sup. Multi-output Manifold Regression	Partial	Low	Unknown	5

which provide weak supervision. In this dissertation, we investigate how the manifold structure of image sets can be learned using weak supervision to solve a variety of image analysis problems.

1.2 Overview of Research

Given the feature representation of an image set, $\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, \cdots, \mathbf{x}_{\mathrm{N}}]^\intercal$, where $\mathbf{x}_i \in \mathcal{R}^{\mathrm{D}}$ corresponds to the image feature representation of image i, we assume that the images are random samples from a manifold $\mathcal{M}$, embedded in the ambient space, $\mathcal{R}^{\mathrm{D}}$. Let $\mathbf{Y} = [\mathbf{y}_1, \mathbf{y}_2, \cdots, \mathbf{y}_{\mathrm{N}}]^\intercal$ be the image labels, where $\mathbf{y}_i \in \{\mathcal{R}^{\mathrm{D}_y}, \varnothing\}$. When $\mathbf{y}_i = \varnothing$, it means the image, $\mathbf{x}_i$, is unlabeled.

Based on the amount and the quality of provided labels and whether prior knowledge on the type of image variation is available, we investigate three different variants of weakly supervised learning on image manifolds. Table 1.1 organizes these problems based on the level of supervision available.

1.2.1 Weakly Supervised Manifold Factorization

Factorization of image variation, i.e., separating different sources of change underlying an image set, is routinely performed by human perceptual systems. For

example, characters can be recognized when written in an unfamiliar style, and the color of an object is perceived as the same under different illumination conditions. In our research, we focus on the situation where two types of latent factors exist in the data: (1) variation of interest which correlates with the provided image labels, and (2) auxiliary variation with a known transformation model (affine, deformable, etc.). The goal is to remove the unwanted auxiliary variation. This problem is weakly supervised in the sense that image labels only explain a part of the intrinsic parameters of a manifold. Figure 1.8 shows a graphical illustration of the problem. The goal is to remove unwanted auxiliary variation from the image set, $\mathbf{X}$, given a proxy of the variation of interest, $\mathbf{Y}$.

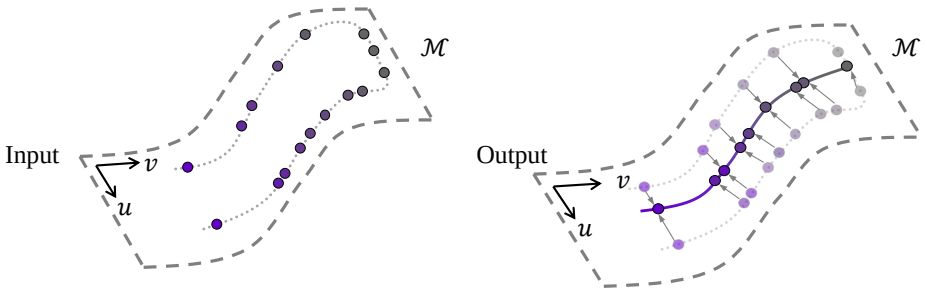


Figure 1.8: Each point represents an input data point and color indicates label values. In this example, the direction represented by $\mathbf{v}$ is the variation of interest and changes in $\mathbf{u}$ are unwanted. The goal is to learn a lower-dimensional manifold parametrized by $\mathbf{v}$ only.

1.2.2 Robust Manifold Regression

In the same way that clustering is a natural tool for classification problems, manifolds provide a natural model for regression problems. We consider the case when the provided image labels are highly corrupted. The goal is to learn a function using the corrupted labels, such that noise-free labels are estimated by mapping each image

using the learned function. Formally, given the image set, $\mathbf{X}$, and the noisy labels, $\mathbf{Y}$, robust manifold regression aims to learn a function, $f : \mathcal{M} \to \mathcal{R}^{\mathrm{D}_y}$, which is defined on the manifold and maps to the D_y-dimensional output space. Figure 1.9 shows that the image manifold structure provides regularization for the labels, as the ideal image labels should be smoothly varying on the manifold.

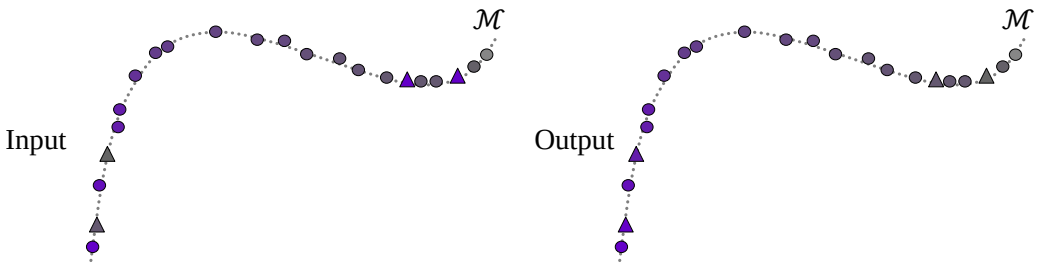


Figure 1.9: Illustration of robust manifold regression. Triangles indicate data points originally associated with corrupted label values.

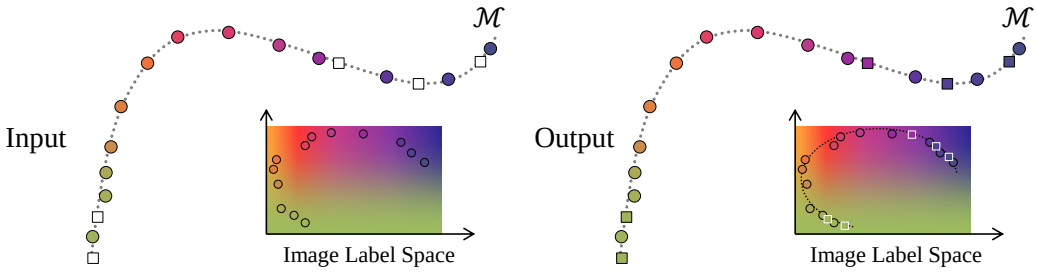


Figure 1.10: Illustration of semi-supervised multi-output manifold regression using a partially labeled toy data set with 2D labels. Square points indicate originally unlabeled data.

1.2.3 Semi-supervised Multi-output Manifold Regression

Compared with classification and regression, there is much less work on multi-output prediction. We consider the semi-supervised setting, where only some of the images are labeled. Given the image set, $\mathbf{X}$, and the associated labels, $\mathbf{Y}$, we assume

there is an underlying low-dimensional structure in $\mathcal{R}^{D_y}$. The goal is to learn the manifold function, $f : \mathcal{M} \rightarrow \mathcal{R}^{D_y}$. Figure 1.10 shows that both the image manifold structure (constructed from labeled and unlabeled data) and the underlying structure of image labels will be utilized to learn the function.

1.3 Dissertation Outline

This dissertation is organized as follows. In Chapter 2, we review related literature and highlight the difference of our problems from previous work. The three variants of weakly supervised learning on image manifolds are addressed individually from Chapter 3 to Chapter 5. To solve each problem, a general framework is provided first, and then an algorithm is proposed according to the specific application domain. All of the algorithms are thoroughly evaluated with multiple experiments on both synthetic and real-world data sets. In Chapter 6, we provide a conclusion for this dissertation and point out future directions for improvement.

CHAPTER 2: RELATED WORK

To make the inference of high-dimensional images tractable, a data model is usually employed to capture the latent structure of images. Many generic data modeling methods have been applied to computer vision problems. In this dissertation, we are specifically interested in data models that can encode the underlying relationships of images according to their underlying factors of change. We then take advantage of the semantically meaningful relationships between images to regularize the function learning of conceptually related image labels for our research problems.

2.1 Generic Data Modeling Methods

A widely used data modeling method is parametric curve fitting, which assumes that the distribution of points in a data set can be approximated by a polynomial curve [10]. Usually, the form of the polynomial is provided and the learning algorithm will estimate the optimal coefficient for each term in the polynomial. However, image sets have complicated latent structures embedded in high-dimensional spaces, which makes it difficult to choose the most suitable polynomial model. In addition, the parameters of the polynomial curve are derived from a high-dimensional feature space, which poses serious challenges to learning the model without overfitting.

Kernel density estimators (KDE) [65, 46] provide a non-parametric approach to statistically estimate the underlying data distribution. In some applications (such

as background intensity modeling [31]), having the probabilistic distribution of the data is sufficient for the task. However, many computer vision problems do not focus on the global distribution of a data set, but investigate the relationships between images. These relationships can show how certain properties of the real world change as an image changes. The output of KDE does not encode any local changes between images and can not provide a principled way to model the interrelationships between images.

With the development of compressed sensing theory [30], sparse data models have been increasingly applied in computer vision [62, 109]. In sparse models, a signal or a data point is represented as the linear combination of a few basis drawn from an over-complete dictionary. Although sparse models provide a compact and robust representation for each image, they do not parametrize changes between images. Therefore, similar to KDE, sparse models are not suitable for situations where there is a need to model image change with respect to a few underlying factors.

A large body of work has been proposed on manifold based models [93], which aim to represent high-dimensional points and their interrelationships using a few perceptually meaningful factors. The manifold assumption is generic and can be applied to many problems in computer vision, as most image sets can be viewed as data points sampled from an underlying manifold embedded in a high-dimensional feature space. In addition, manifold based methods are usually non-parametric, meaning they tend to be less sensitive to data dimensionality than a parametric model such as parametric curve fitting.

2.2 Image Manifold Representation

The early work that investigated computational models for manifold representation often focused on devising nonlinear dimensionality reduction techniques. For example, low-dimensional coordinates are computed that maintain the local distributions of points on the manifold [80, 11], or optimize a global objective function (e.g., reconstruction error of geodesic distances [90], data variance in embedded space [105]). Recently, there has been increased interest in formulating computer vision problems on image manifolds. For example, for the problem of non-rigid object segmentation, a shape manifold is used to facilitate efficient searching of the optimal segmentation in the shape space [68]; face pose is estimated by aligning the manifold of local image patches and the manifold of 3D shape patches [103]; in a top-down (rigid initialization and nonrigid refinement) segmentation framework, sparse manifolds are used to reduce the search space of rigid transformations [67]. In addition, manifold models have been the basis for image denoising [61, 94, 37], deformable registration [115], and action recognition [1, 36] among others.

However, many of the previous manifold modeling methods estimate the underlying manifold in an unsupervised way, meaning images are the only input for estimating the manifold. Unsupervised methods tend to work well if the images densely sample the underlying manifold. However, for real-world applications, where imaging noise and sparse sampling in high-dimensional spaces complicate the problem, estimating the manifold structure is non-trivial. In this situation, image labels that provide information about the intrinsic parameters of a manifold can help to constrain the

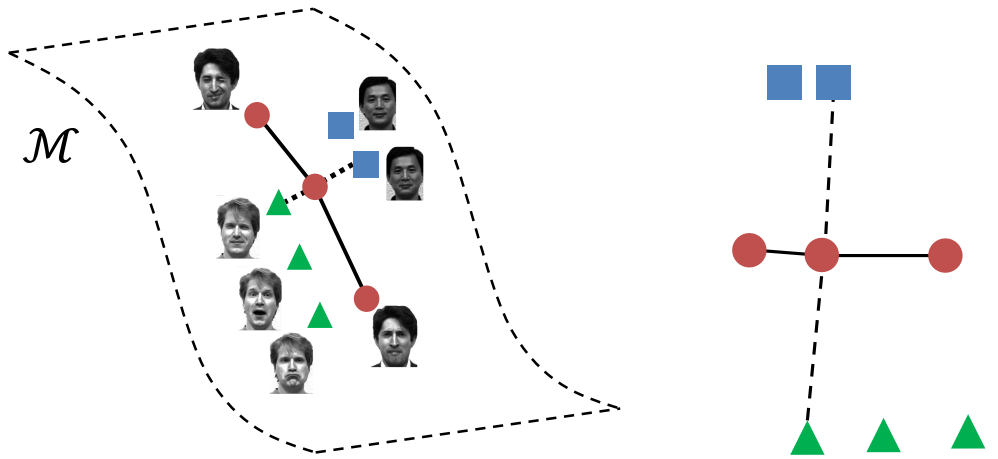
problem.

2.2.1 Supervised Learning on Image Manifolds

When relevant information (often in the form of data labels) is available, it can be used to regularize the underlying manifold structure for specific tasks (e.g., finding the most discriminant features [14, 25], discarding unwanted factors of variation [9, 59, 114], and image denoising [110, 112]). We use the following two examples to demonstrate how image labels can be incorporated for supervised learning on image manifolds.

First, a supervised method for nonlinear dimensionality reduction assumes that images lie on a manifold parametrized by multiple latent factors [14]. Similar to many other manifold learning methods, the underlying manifold structure is estimated using the neighborhood graph on images. However, different from manifold learning (where the goal is to preserve the relationships of images), the method finds the most discriminative low-dimensional representation for classification tasks. Using the provided image class labels, the neighborhood graph is divided into a within-class subgraph and an inter-class subgraph. The optimal embedding minimizes the distances of nearby points with the same class labels while separating nearby points from different classes. For example, in Figure 2.1, face images of different identities are far apart after dimensionality reduction.

Also, we have applied supervised manifold learning to the problem of image denoising [110, 112]. Existing manifold denoising methods treat noisy images as outliers from an ideal manifold and denoise images by projection onto the manifold. Unsu-



(a) A local neighborhood of images on the manifold. (b) The same images after dimensionality reduction.

Figure 2.1: Illustration of a supervised method for dimensionality reduction [14]. Image class labels (denoted by color and shapes) are used to find the optimal projection. Within-class distances (solid lines) are minimized while inter-class distances (dashed lines) are maximized. After dimensionality reduction, the output is more discriminative (b).

pervised methods [102, 44, 37] usually adopt an iterative approach that alternates between estimating the manifold structure and denoising points. This tends to be sensitive to noise and the high dimensionality of the ambient space. Provided with image labels that correlate with the underlying image variation of interest, supervised manifold denoising directly uses image labels to explicitly parametrize the manifold (Figure 2.2) and estimate denoised images using the optimal reconstruction of images on the learned manifold.

Although supervised image manifold learning often outperforms unsupervised methods in certain tasks, the full image annotations required by supervised learning are usually hard to obtain. With the emergence of large-scale image sets [26], collecting full annotations becomes even more impractical, which hinders the wide use of supervised methods. However, in many situations, a trade-off can be achieved using weak

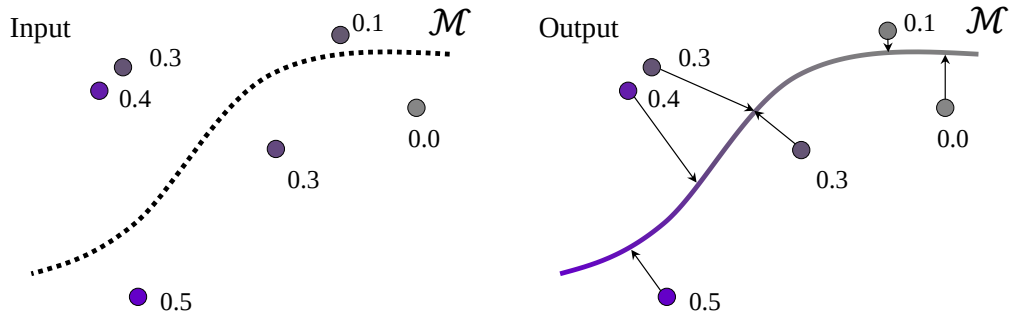


Figure 2.2: Illustration of supervised manifold denoising [110, 112]. Each point represents a noisy image; the label values are indicated by the numbers and the color of each point. Images are denoised by finding their projections onto the underlying manifold parametrized by image labels.

labels which greatly reduce the labeling effort and maintain a certain level of supervision. However, with the prevalence of manifold modeling in computer vision, most of the previous work in image manifold learning has mainly focused on either fully supervised or unsupervised learning. In this dissertation, we study various scenarios in the underserved area of weakly supervised learning on image manifolds.

2.3 Weakly Supervised Learning

Unlike supervised learning, where accurate image labels are provided for each image, weak supervision can be in many forms. Some methods use image labels that are less detailed than the ideal ground truth labels. For example, in object segmentation, instead of providing the exact foreground delineation, a bounding box surrounding the foreground object or even only image-level tagging of the object class is provided [40, 121]. Inaccurate labels have also been used in weakly supervised learning: human actions are recognized from movies using noisy action labels extracted from the associated movie scripts [29]. Many other methods consider cases with partial labels [18, 123]. For example, for dimensionality reduction [15] and hashing [99],

class labels on a small subset of the images are incorporated to preserve semantic similarity of images from the same class. Different from previous work in weakly supervised learning, our research utilizes the manifold structure of images to regularize the problem.

Existing work in weakly supervised learning on image manifolds has primarily considered the semi-supervised setting. Methods in this category usually incorporate partially provided image labels and propagate the labels over the manifold approximated by the neighborhood graph on the images [116, 60, 8, 53]. However, in a broad sense, different situations of weak supervision have not been well studied in this category. Motivated by this observation, our research aims to explore different scenarios of weakly supervised learning on image manifolds.

CHAPTER 3: WEAKLY SUPERVISED MANIFOLD FACTORIZATION

Weakly supervised manifold factorization considers the case where two types of latent factors are present in images: the auxiliary variation and the variation of interest. The goal is to incorporate image labels that are correlated with the variation of interest and learn a factorization model that removes the image changes caused by the auxiliary variation. As shown in Figure 3.1, after removing unwanted image variation, image changes are entirely parametrized by the variation of interest. Weakly supervised manifold factorization corresponds to many problems in computer vision. For example, removing the global image motion due to camera movement produces more visually pleasing output for a video clip captured by a free-hand camera (Figure 3.2(a)), and removing illumination changes in a set of face images makes the image set more suitable for downstream applications, such as face synthesis at new pose angles (Figure 3.2(b)).

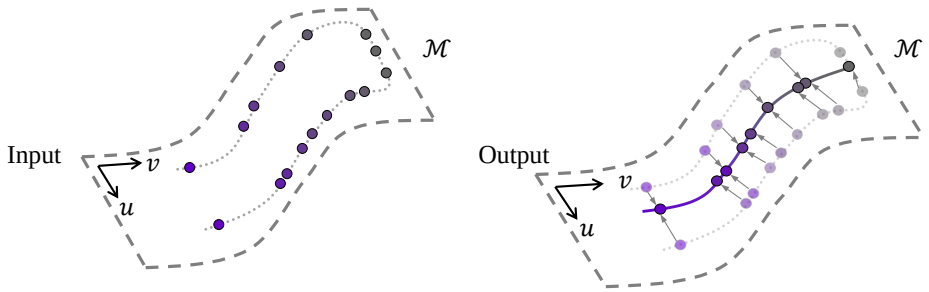


Figure 3.1: Each point represents an input data point and color indicates label values. In this example, the direction represented by **v** is the variation of interest and changes in **u** are unwanted. The goal is to learn a lower-dimensional manifold parametrized by **v** only.

In the supervised setting, the factorization model is learned using image labels on both the auxiliary variation and the variation of interest. However, image appearance changes due to the auxiliary variation are often less structured and more difficult to annotate than change due to the variation of interest. For example, labeling phases of a bird's flapping cycle is easier than providing labels of the sensor induced motion (Figure 3.2(a)), and labeling face pose is more convenient than providing annotations for the illumination variation (Figure 3.2(b)). Weakly supervised manifold factorization incorporates image labels associated with the variation of interest and bypasses the problem of annotating the auxiliary variation in these applications.

(a) A video captured by a free-hand camera containing undesired sensor induced motion

(b) A set of face images with various pose angles containing undesired illumination change

Figure 3.2: Most image sets contain multiple latent factors of change. By removing undesired latent factors, images are more suitable for direct application or downstream analysis.

In this chapter, we present a basic framework for weakly supervised manifold factorization. We use the provided labels to find image pairs that should be similar after removing the unwanted image variation and propose an efficient, keyframe based optimization approach. We demonstrate how this framework can be applied for echocardiogram video stabilization and provide experimental evaluation of the algorithm on

both synthetic and real-world data sets.

3.1 Background

Previous research on image variation factorization has mainly been concerned with separating "style" and "content" factors from images [91, 19, 32, 98]. These approaches use an explicit training stage to learn the factorization model, where the training images can be organized by the provided "style" and "content" labels. Usually, they deal with fully labeled image sets, and at least one of the two factors is discrete. Whereas weakly supervised manifold factorization deals with the case where the latent factors are continuous, and the image labels only explaining part of the latent factors.

In terms of removing irrelevant variation and preserving informative variation, manifold alignment [97, 95, 96] shares a similar goal to weakly supervised manifold factorization. However, manifold alignment methods usually aim to find a common coordinate space for multiple data sets, so that knowledge can be transferred between different data sets. However, in our addressed problem, we focus on removing the irrelevant variation within a single data set.

3.2 Framework

Given input images $\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, \cdots, \mathbf{x}_N]^\intercal$, we assume that the underlying manifold, $\mathcal{M}$ is entirely parametrized by the variation of interest, $\mathbf{v}$, and the auxiliary variation, $\mathbf{u}$. The nonlinear relationship between the manifold parameters and the images in ambient space can be represented as a function: $g : \mathcal{R}^{D_u} \times \mathcal{R}^{D_v} \rightarrow \mathcal{M}$. Factoring the variation of $\mathbf{X}$ into component factors is nontrivial and under-constrained. However

we consider the version of this problem with weak labels, where image labels, $\mathbf{Y}$, that approximate the variation of interest are provided and serve as an additional constraint.

Taking any pair of images in $\mathbf{X}$, identifying the contribution of each source of variation is under-constrained. However, with fixed $\mathbf{v}$, the difference between two images, $g(\mathbf{u}_a, \mathbf{v})$ and $g(\mathbf{u}_b, \mathbf{v})$, is entirely from the auxiliary variation. Applying the brightness constancy constraint gives us:

$$T(g(\mathbf{u}_a, \mathbf{v}); \mathbf{u}_a) = T(g(\mathbf{u}_b, \mathbf{v}); \mathbf{u}_b) \tag{3.1}$$

where $T(\cdot; \mathbf{u})$ is a image transformation parametrized by $\mathbf{u}$; for an image $g(\mathbf{u}, \mathbf{v})$, $T(g(\mathbf{u}, \mathbf{v}); \mathbf{u})$ transforms the image to a standard parameter setting of auxiliary variation, $g(\mathbf{u}_0, \mathbf{v})$. Given discrete samples of the image manifold, the relationship in Equation 3.1 can be made between each pair of images with similar $\mathbf{v}$. Since the image labels approximate the variation of interest, we can use the similarity of image labels, $\kappa(\|\mathbf{y}_i - \mathbf{y}_j\|_2)$ to estimate the similarity of $\mathbf{v}_i$ and $\mathbf{v}_j$, , where $\|\cdot\|_2$ is the $L2$-norm, and $\kappa(\cdot)$ is the radial basis kernel. Extending from image pairs to the entire image set, we have:

$$\min_{\mathbf{U}} \sum_{i=1}^{\mathrm{N}} \sum_{j=1}^{\mathrm{N}} \kappa(\|\mathbf{y}_i - \mathbf{y}_j\|_2) \phi\left(T(\mathbf{x}_i; \mathbf{u}_i), T(\mathbf{x}_j; \mathbf{u}_j)\right) + \lambda_{\mathrm{a}} P(\mathbf{U}) \tag{3.2}$$

where $\mathbf{U} = [\mathbf{u}_1, \cdots, \mathbf{u}_{\mathrm{N}}]^\intercal$, $\phi(\cdot)$ is an image dissimilarity measure, and $P(\cdot)$ is a regularization term for any additional constraint on $\mathbf{U}$ (e.g., temporal constraint, in the case of videos).

An issue with the above formulation is that evaluation of all-pair image matching

(even for simple transformation models) is computationally expensive, and at the same time, only a small subset of the all-pair computation (image pairs with very similar image labels) contributes to the minimization. We use an efficient strategy to avoid this all-pair computation: a small subset of the image set is selected and serves as the set of reference images to align the entire image set. Formally, given the set of images, $\mathcal{I} = \{\mathbf{x}_i\}_{i=1}^{\mathrm{N}}$, and the associated labels, $\mathbf{Y}$, we select a subset $\mathcal{L}$ from $\mathcal{I}$ as the reference images, namely the *keyframes*, and solve a modified version of Equation 3.2:

$$\min_{\mathbf{U}} \sum_{i} \phi\left(T(\mathbf{x}_i; \mathbf{u}_i), T(\mathbf{x}_{\gamma(i)}; \mathbf{u}_{\gamma(i)})\right) + \lambda_{\mathrm{a}} P(\mathbf{U}) \tag{3.3}$$

where $\gamma(i) \in \mathcal{L}$ denotes the index (in the original image set $\mathcal{I}$) of the selected keyframe for image $\mathbf{x}_i$.

3.3 Phase-aware Echocardiogram Stabilization using Key Frames

2D echocardiography is a ubiquitous approach for the real-time, noninvasive analysis of heart function. With the increased use of portable ultrasound devices in critical care settings, methods for automated echocardiogram analysis are increasingly relevant for situations when cardiologists are not available for diagnosis. There has been much work in automated cardiac motion analysis from echocardiograms, including left ventricle segmentation and tracking [71, 81, 120, 58], statistical modeling of atlases [38, 42], and quantitative assessment of cardiac motion [28, 87]. An assumption implicit in most of these algorithms is that observed motion is primarily due to cardiac motion (potentially corrupted by noise). However, real-world echocardiograms show variations due to a variety of auxiliary causes, including (1) patient breathing and (2)

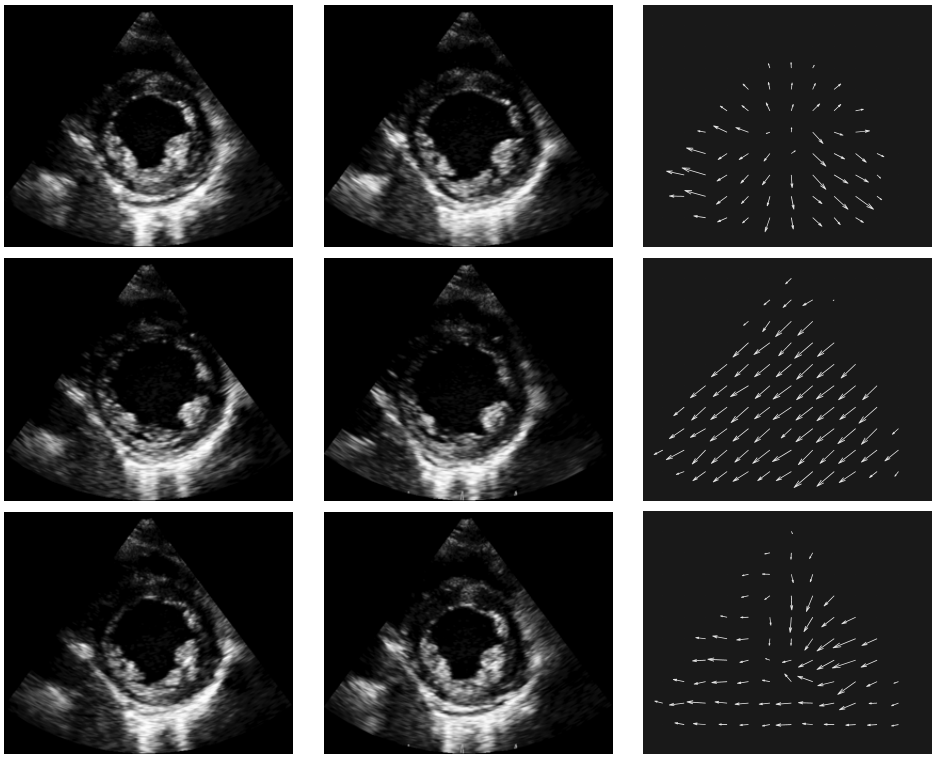


Figure 3.3: Each row shows a pair of echocardiogram frames, and the image motion is represented as vector fields. Most alignment methods deal with either cardiac motion (top, deformable) or sensor motion (middle, approximately rigid). However, most pairs of echocardiogram frames (bottom) vary due to both.

movement of the handheld transducer relative to patient. Both of these causes can appear as rotational, translational, and out-of-plane motion, which, similar to [6], we model as approximately rigid motion.

Figure 3.3 shows selected frames from a transthoracic echocardiogram obtained by a trained ultrasound technician using a handheld transducer from a session consisting of six heartbeats, lasting roughly five seconds. Each row shows a pair of echocardiogram frames and the image motion represented as vector fields. The top row shows an end-systolic frame and an end-diastolic frame from the same cycle. The middle row shows two end-diastolic frames from different cycles. While the pattern of motion differs, the overall magnitude in each case is similar. That is, auxiliary motion can be just as

significant as the informative cardiac motion. The third row shows two frames from different phases and cycles where a composite of both sources can be observed.

In this chapter, we propose a solution to the problem of video stabilization for freehand 2D echocardiography based on the framework of weakly supervised image factorization. We assume the images are random samples from a manifold parametrized by cardiac motion (variation of interest, $\mathbf{v}$) and relative sensor motion (auxiliary motion, $\mathbf{u}$) and seek to compensate for the auxiliary motion without affecting motion due to cardiac cycles.

3.3.1 Related Work on Video Stabilization

Most of the work in video stabilization has focused on natural scenes captured from consumer handheld cameras, where the goal is to produce visually pleasing output from video corrupted by undesired jitter motion [13, 66]. However, many of the underlying assumptions of these methods do not hold for echocardiograms. Many of these approaches assume that most of the image area contains non-moving objects, so the images can be stabilized by tracking image features [57] or estimating the global motion between consecutive frames [63]. However, for typical freehand echocardiography, cardiac structures occupy most of the image area, and the image changes due to cardiac motion are significant, so the estimated global motion is unlikely to correspond only to sensor motion.

Video stabilization is related to the problem of group-wise image registration. Compensating for the motion caused by the sensor can be viewed as registering the constituent frames onto a common coordinate frame where only deformable cardiac mo-

tion is preserved. Most of the work in group-wise image registration with deformable objects seeks to find the optimal parameters of a prescribed deformable motion model with respect to some image similarity measure [85, 21, 86]. This differs from our problem in that we do not aim to parametrically estimate the total deformable motion, but rather factor nuisance sensor-caused motion. Perhaps more closely related is group-wise rigid alignment. Most of these approaches rely on the assumption that when images are optimally aligned, they are the most similar to a mean image. A variety of alignment techniques have been employed for this problem, including low rank decomposition [74], least squared difference [23, 22], and entropy minimization [56]. For echocardiograms, there are significant non-rigid deformations over the course of a cardiac cycle, and, as we demonstrate in Chapter 3.3.4, a single reference or mean image does not work well for alignment. Our approach is to efficiently learn a set of keyframes for stabilizing the video.

There are two recent approaches for group-wise non-rigid image alignment that share a similar model to our work. When considered as points in a high-dimensional space, a set of images related by a few underlying degrees of freedom, sensor motion and cardiac motion in our case, lie on or near a low-dimensional manifold embedded in this image space. Rather than considering the sequential relationship of frames inherent to video, the manifold model instead considers image-image similarities across the entire video. The manifold structure is approximated as a graph where each node is an image connected to its most similar neighbors [115]. All the images are then registered to the population center (the image with the closest geodesic distances to all other points) by graph shrinkage. This differs from our approach in the same way

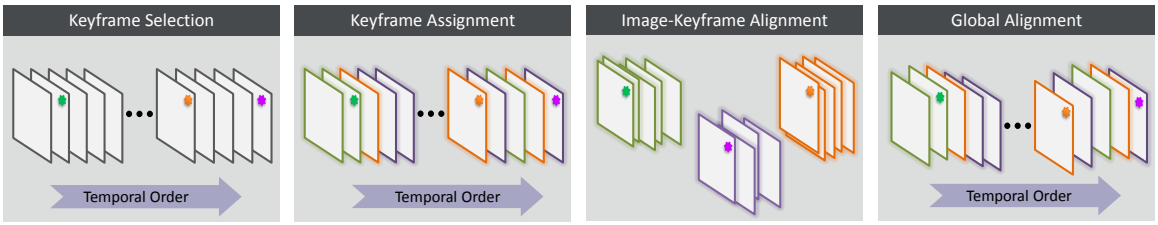


Figure 3.4: Overview of the proposed video alignment algorithm. The algorithm begins by selecting keyframe images and assigning each image to the most suitable keyframe for local alignment. Global alignment incorporates both image-keyframe alignment and temporal smoothness.

as other non-rigid alignment approaches that deform images, but shares a similar manifold representation of video. A previous work of ours solves a similar problem and also employs keyframe based alignment to avoid pairwise registration [111]. But the approach proposed in [111] uses a greedy method to select keyframes in an empirical way; whereas our method presented in this chapter formulates the problem as a novel graph optimization problem and provides a theoretical bound for the proposed efficient keyframe selection strategy.

3.3.2 Method

Even for simple transformation models and similarity metrics, Equation 3.3 leads to a nonlinear, non-convex optimization. We propose an approach to efficiently approximate the solution to Equation 3.3. Figure 3.4 depicts each component of our video stabilization algorithm: keyframe selection, image-keyframe alignment, and global refinement.

3.3.2.1 Keyframe Selection

Each image will be assigned to one of n_k keyframes for pairwise alignment. When $n_\mathrm{k} = 1$, all images are aligned to a single reference. As n_k increases, more keyframes

are selected, and images are aligned to the keyframe with the most similar label, but at increasing computational cost. The number of keyframes n_{k} can act as a free parameter that controls the trade-off between alignment accuracy and computational cost. In our experiments, we show how n_{k} can be specified implicitly by computing the marginal gain of adding an additional keyframe and terminating if the gain falls below a threshold. However, here we leave n_{k} as a free parameter for clarity.

First, we compute the compatibility of two images, $\mathbf{x}_i$ and $\mathbf{x}_j$, for alignment. Ideally, an image should be aligned to another image that differs primarily due to auxiliary motion, i.e., the deformable shape represented by the two images are very similar. Given the compatibility between image $\mathbf{x}_i$ and image $\mathbf{x}_j$, $w_{ij} = \kappa(\|\mathbf{y}_i - \mathbf{y}_j\|_2)$, we have the kernel matrix between the image labels of all pairs of images in $\mathcal{I}$. We construct a fully-connected, undirected graph with each image as an vertex, and w_{ij} as the edge weight between the i-th and j-th vertexes. The optimal set of keyframes would be the subset that maximizes the sum of pairwise compatibility, where each image is matched to a keyframe. Formally, given a set of keyframes, $\mathcal{L} \subseteq \mathcal{I}$, the objective function in this graph optimization problem is:

$$F(\mathcal{L}) = \sum_{i=1}^{\mathrm{N}} \max_{\mathbf{x}_j \in \mathcal{L}} w_{i,j} \tag{3.4}$$

where the max term ensures that each image will be aligned to a single keyframe. Keyframe selection becomes a discrete optimization of the form:

$$\max_{|\mathcal{L}| \leq n_{\mathrm{k}}} \quad F(\mathcal{L}) \tag{3.5}$$

This combinatorial optimization problem is a variant of the generalized maximum cov-

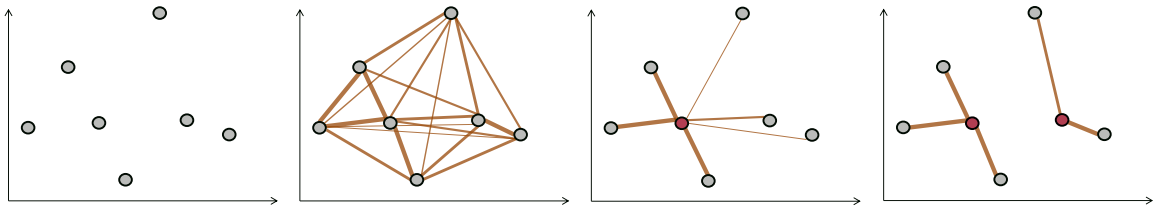


Figure 3.5: (left) Each point represents an input image plotted by the associated label values. (second) Vertexes are linked by edges weighted by image compatibility. (third and last) Two keyframes (red) are selected sequentially by applying the greedy algorithm twice, and the edges represent the assignment of each image to the current keyframe(s).

erage problem (GMC), which is NP-hard. Fortunately, this function is monotonically non-decreasing with submodular structure [20], which allows us to take advantage of the following theorem to employ a constant-factor approximation algorithm.

Theorem 3.3.1. [70] *Given a non-decreasing submodular function F, $F(\emptyset) = 0$, the greedy maximization algorithm returns $\mathcal{A}_{greedy}$, and $F(\mathcal{A}_{greedy}) \geq (1 - \frac{1}{e}) \max_{|\mathcal{A}| \leq n_k} F(\mathcal{A})$.*

For a toy example with 2D metadata, Figure 3.5 presents an overview of keyframe selection that iteratively selects the keyframe with the maximum weight gain, until n_k keyframes are selected. We denote the set of keyframes obtained as $\mathcal{L}_\mathrm{g}$.

Once the keyframe images are selected, the next step is to align each image, $\mathbf{x}_i$, to the corresponding keyframe, $\mathbf{x}_{\gamma(i)}$. This subproblem (matching a keyframe to the subset of paired frames) is an instance of image-image alignment. Let $\boldsymbol{\theta}_i$ represent the transformation parameters that align image $\mathbf{x}_i$ to the corresponding keyframe, $\mathbf{x}_{\gamma(i)}$. Even for this single-reference subproblem, solving for $\boldsymbol{\theta}_i$ leads to a nonlinear, non-convex optimization. We use Bayesian optimization to solve for the image-keyframe alignment parameters, which has shown to be more efficient and accurate than grid and other random searches [84]. The obtained image-keyframe alignment parameters,

$\boldsymbol{\Theta} = [\boldsymbol{\theta}_1, \boldsymbol{\theta}_2, \cdots, \boldsymbol{\theta}_{\mathrm{N}}]^\intercal$ provide an initial alignment from each image to a keyframe. In the next chapter, we show how these values are used to solve for the global alignment parameters.

3.3.2.2 Global Alignment

Given image-keyframe alignment parameters, $\boldsymbol{\Theta}$, we want to solve for the global parameters $\mathbf{U} = [\mathbf{u}_1, \ldots, \mathbf{u}_{\mathrm{N}}]^\intercal$ that align all of the frames with respect to a common coordinate frame. The desired global alignment parameters of the i-th image and its assigned keyframe, $\mathbf{u}_i$ and $\mathbf{u}_{\gamma(i)}$, should agree with the image-keyframe alignment parameters, $\boldsymbol{\theta}_i$. This leads to the constraint on $\mathbf{U}$: $T(\mathbf{x}_{\gamma(i)}, \mathbf{u}_{\gamma(i)}) = T(T(\mathbf{x}_{\gamma(i)}, \boldsymbol{\theta}_i), \mathbf{u}_i)$. That is, keyframe $\mathbf{x}_{\gamma(i)}$ and image $\mathbf{x}_i$ (substituted by $T(\mathbf{x}_{\gamma(i)}, \boldsymbol{\theta}_i)$ using the image-keyframe alignment constraint), should be optimally aligned using $\mathbf{u}_{\gamma(i)}$ and $\mathbf{u}_i$. In the case of the approximately rigid auxiliary motion, this leads to: $\mathbf{u}_{\gamma(i)} - \mathbf{u}_i = \boldsymbol{\theta}_i$. Enforcing this constraint on $\mathbf{U}$ approximates a minimization of the first term in Equation 3.3. For the case of video stabilization, the second term $P(\cdot)$ can enforce temporal smoothness, $\sum_i \left|\frac{\partial \mathbf{u}_i}{\partial i}\right|^2$. Using central differences to approximate the first derivative in the temporal smoothness term, we have the following objective function for global alignment:

$$\begin{aligned} \underset{\mathbf{U}}{\operatorname{argmin}} \quad & \|\mathbf{D}\mathbf{U} - \boldsymbol{\Theta}\|_F^2 + \lambda_{\mathrm{a}} \|\mathbf{L}\mathbf{U}\|_F^2 \\ \text{subject to} \quad & \mathbf{u}_1 = 0 \end{aligned} \tag{3.6}$$

where $\mathbf{D}$ is a N $\times$ N matrix consisting of -1s along the diagonal and 1s on the $\gamma(i)$-th element of the i-th row, and 0s everywhere else; and $\mathbf{L}$ is a N $\times$ N Laplacian

matrix obtained from the graph of connecting temporally adjacent images. The first term ensures that the global parameters agree with the image-keyframe alignment parameters, and the second term ensures temporal smoothness. The constraint, $\mathbf{u}_1 = 0$, removes translational ambiguity of the global coordinate. Let $\mathbf{U} = \tilde{\mathbf{I}}\mathbf{\Gamma}$, where $\tilde{\mathbf{I}}$ is a N $\times$ N matrix containing 1s along the diagonal except for the first element, and 0s everywhere else, and $\mathbf{\Gamma} \in \mathcal{R}^{\mathrm{N}\times\mathrm{D}_u}$. Substitute $\mathbf{\Gamma}$ into Equation 3.6, and we are left with the following unconstrained convex quadratic minimization problem:

$$\mathbf{\Gamma}^* = \underset{\mathbf{\Gamma}}{\operatorname{argmin}} \|\mathbf{D}\tilde{\mathbf{I}}\mathbf{\Gamma} - \mathbf{\Theta}\|_F^2 + \lambda_{\mathrm{a}} \|\mathbf{L}\tilde{\mathbf{I}}\mathbf{\Gamma}\|_F^2 \tag{3.7}$$

The above equation can be solved efficiently by solving the linear system $\mathbf{A}\mathbf{\Gamma} = \mathbf{B}$ using the conjugate gradient method, where $\mathbf{A} = (\mathbf{D}\tilde{\mathbf{I}})^\intercal(\mathbf{D}\tilde{\mathbf{I}}) + \lambda_{\mathrm{a}}(\mathbf{L}\tilde{\mathbf{I}})^\intercal(\mathbf{L}\tilde{\mathbf{I}})$, and $\mathbf{B} = (\mathbf{D}\tilde{\mathbf{I}})^\intercal\mathbf{\Theta}$.

3.3.2.3 Algorithm

Algorithm 1 provides pseudocode for the proposed *Video Stabilization using Phase-Aware Keyframes* (`VSPAK`), broken down into: keyframe selection (lines 1 – 5), image-keyframe alignment (lines 6 – 7), and global alignment (lines 8 – 9).

3.3.3 Experimental Evaluation

In this chapter, we quantitatively evaluate our approach for video stabilization on synthetic videos (with known ground truth) and compare the results to related methods. For all the repeated experiments in Chapter 3.3.3 and Chapter 3.3.4, we performed pairwise comparisons with a two-sample t-test with a significance value of $\alpha = .05$.

Algorithm 1 VSPAK

Input: images, $\mathcal{I} = \{\mathbf{x}_i\}_{i=1}^{\mathrm{N}}$; metadata, $\mathbf{Y} = [\mathbf{y}_1, \mathbf{y}_2, \cdots, \mathbf{y}_{\mathrm{N}}]^\intercal$; number of keyframes, n_{k}; trade-off coefficient, λ_{a}.

Output: alignment parameters, $\mathbf{U}$.

1: Compute pairwise suitability, $w_{ij} = \kappa(\|\mathbf{y}_i - \mathbf{y}_j\|_2)$
2: Initialize the keyframe set, $\mathcal{L}_{\mathrm{g}} \leftarrow \emptyset$
3: **while** $|\mathcal{L}_{\mathrm{g}}| \leq n_{\mathrm{k}}$ **do**
4: $\quad \mathbf{x}_j \leftarrow \underset{\mathbf{x} \in \mathcal{I} - \mathcal{L}_{\mathrm{g}}}{\operatorname{argmax}} F(\mathcal{L}_{\mathrm{g}} + \mathbf{x}) - F(\mathcal{L}_{\mathrm{g}})$
5: $\quad \mathcal{L}_{\mathrm{g}} \leftarrow \mathcal{L}_{\mathrm{g}} \cup \mathbf{x}_j$
6: **for all** $\mathbf{x}_i$ **do**
7: $\quad$ Compute image-keyframe alignment, $\boldsymbol{\theta}_i = \underset{\boldsymbol{\theta}}{\operatorname{argmin}}\ \phi\left(T(\mathbf{x}_i; \boldsymbol{\theta}), \mathbf{x}_{\gamma(i)}\right)$
8: Solve for $\boldsymbol{\Gamma}^*$ (Eq. 3.7)
9: Compute alignment parameters, $\mathbf{U} = \tilde{\mathbf{I}}\boldsymbol{\Gamma}^*$

The synthetic video set was constructed by applying a parametric non-rigid warp (deformable motion) to an initial star-like shape and translating the shape along an arbitrary path (rigid motion) with dynamic range of $[-20, 20]$ in pixel units in both vertical and horizontal directions. The magnitude of the deformable motion is controlled by a metadata parameter between 0 and 1. To add realistic speckle noise to the synthetic video, we use the Field II ultrasound simulation toolbox [50, 51]. For the synthetic video, the foreground (star shape) and background intensities are modeled parametrically to generate speckle noise. The energy strengths of scatters in the background are drawn from a zero-mean Gaussian distribution with standard deviation of 1, and the energy strengths of foreground scatters are drawn from a zero-mean Gaussian distribution with standard deviation of 0.1. In each frame, $10,000$ scatters are randomly placed. Figure 3.6 shows three sample frames from this data set.

The algorithm was implemented in Matlab on a standard desktop computer. For the radial basis function used for metadata similarity, $\kappa(\cdot)$, the kernel width is selected

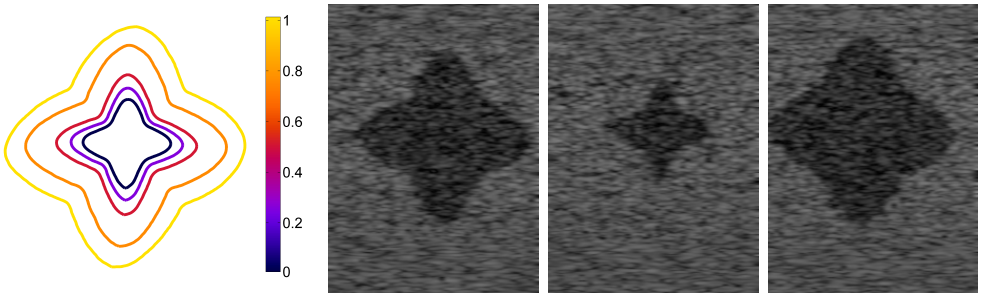


Figure 3.6: The synthetic video data contains simultaneous deformable and rigid motion with imaging noise added.

as the mean pairwise Euclidean distances between the metadata of all images in a video. The number of keyframe images was determined based on the marginal gain of adding an additional keyframe; iterative keyframe selection was terminated when the ratio of the k-th iteration and the first iteration falls below 10^{-2}. The trade-off parameter, $\lambda_{\mathrm{a}} = 1$. For image-keyframe alignment, we used an implementation of Bayesian optimization [35] with 20 iterations.

3.3.3.1 Alignment Accuracy

In this chapter, we evaluate the alignment accuracy using synthetic videos of 300 frames with randomized translation trajectories and noise patterns. The proposed algorithm is compared against the following rigid alignment algorithms. For each method, the free parameters were selected that produced the best results.

- `BASE`: baseline approach where each frame is aligned to a reference image (randomly selected from the input video). To compute the alignment parameters for each image, the reference image is transformed over a grid sampling of the parameter space, and the optimal alignment parameter corresponds with the nearest transformed reference image.

- `Healy2007` [41]: similar to `BASE`, but alignment is based on random projections of the manifold formed by the transformed versions of the reference image and 1-NN search in the projected space. The dimension of the projected space is set to 400.

- `Wu2014` [111]: a phase-aware alignment algorithm based on keyframe image selection and random projections. This method selects keyframe images sequentially and does not include a separate global refinement step. The threshold to determine phase-similar images is $\sim$10% of the dynamic range of the phase metadata. The dimension of the projected space is set to 400.

- `VSPAK`: the proposed method.

For `BASE`, `Healy2007`, and `Wu2014`, the sampling space for alignment parameters is $[-20, 20]$ with a spacing of 1 pixel unit in both directions. The reference image for `BASE` and `Healy2007`, and the seed keyframe image of `Wu2014` are randomized. For all methods, alignment accuracy is computed as the root-mean-square error (RMSE) in pixel distances between the returned alignment parameters and the ground truth location. Figure 3.7 shows the mean RMSE across repeated experiments. Overall, the single reference based methods produce the worst performance, and our algorithm, `VSPAK`, outperforms the other methods by a wide margin. The difference in alignment accuracy between `BASE` and `Healy2007` was not significant ($p = 0.397$), and the improvement in alignment for `VSPAK` is statistically significant compared to `Wu2014` ($p = 0.0002$), `Healy2007` ($p = 0.0014$), and `BASE` ($p = 0.0074$).

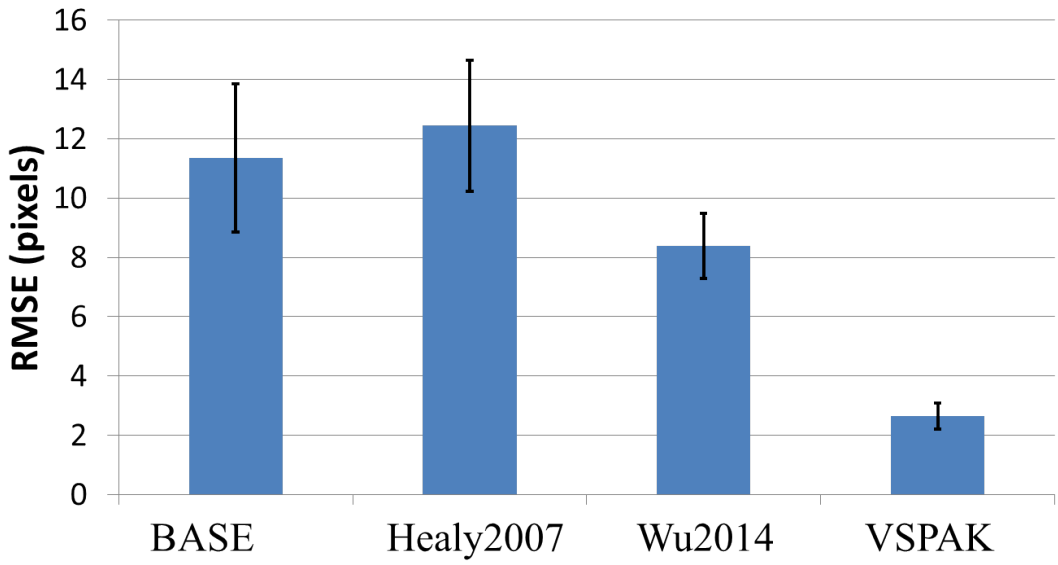


Figure 3.7: Mean RMSE results on synthetic videos in pixel units. Error bars represent standard error.

3.3.4 Echocardiogram Applications

In this chapter, we demonstrate how our video stabilization method can serve as a preprocessing step to improve common automated echocardiogram algorithms. The data for these experiments consists of apical four-chamber (A4C) and parasternal short axis (PSSX) echocardiograms collected from patients in a clinical setting using a Philips CX50 Ultrasound System, operating at 33Hz. Each of the obtained videos contains roughly 6 to 10 heartbeats. To extract image labels that are related with the cardiac movement, we incorporate ECG signals that are collected alongside the echocardiogram. An example ECG is shown in the top row of Figure 3.8. Key points are located in the signal to separate systole and diastole phases [119]. The heart phase parameter is interpolated linearly within each phase for each frame and then projected to a 2D unit circle composed of a systole semi-circle and a diastole semi-

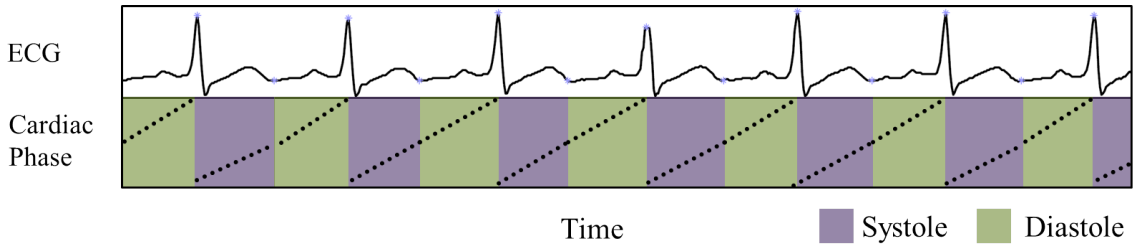


Figure 3.8: Heart phase is inferred for systole and diastole by interpolating between key points in the ECG data.

circle to generate the image labels used for alignment. The resulting image label is a 2D signal and changes along the unit circle alternating between phases of cardiac cycles.

For `BASE`, `Healy2007`, and `Wu2014`, the sampling space for alignment is $[-10, 10]$, and the spacing is 1 pixel unit in both directions. The implementation details for each algorithm are the same as in Chapter 3.3.3.

3.3.4.1 Left Ventricle Segmentation

Left ventricle segmentation is the most common application in cardiac image analysis, with many different algorithms designed for this problem. We applied a recent method [113], which uses an adaptive diffusion flow active-contour model. Except for the first frame (initialized manually), the contour for each frame is initialized using the final contour from the previous frame. For video, when phase information is available, one approach is to re-order the video by phase rather than temporally, prior to sequential segmentation. In these experiments, we show the performance of the segmentation algorithm with and without video stabilization as a preprocessing step.

Figure 3.9 shows the segmentation results on four sample echocardiogram frames

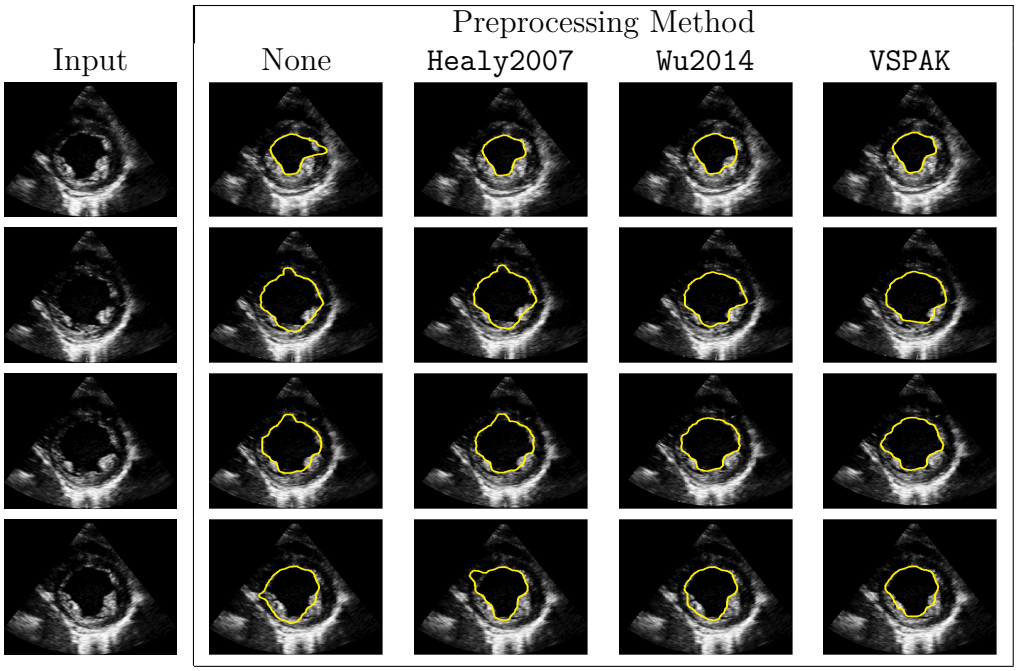


Figure 3.9: Results using a segmentation algorithm [113] on echocardiograms with various video preprocessing methods.

using various preprocessing schemes. Results on the original data (second column) show inaccurate segmentation or boundary leakage on multiple frames. This suggests that the motion between frames may not be smooth enough for the detected boundary from one frame to serve as initialization for the next. Alignment using the single-reference method `Healy2007` (third column) brings insignificant improvement in segmentation. The multi-reference method, `Wu2014` (fourth column) improves the segmentation results compared to the unaligned case, but segmentation errors are still present on the first and the fourth frames. When our method, `VSPAK`, is used as a preprocessing step (last column), the resulting segmentation from the same segmentation algorithm on the same data are noticeably better than the other approaches. This suggests that, without any additional modifications, accurate video stabilization can positively impact downstream automated algorithms.

3.3.4.2 Echocardiogram Denoising

Echocardiogram denoising suppresses the speckle patterns commonly exhibited in echocardiogram images, which, in some cases, can be useful features for motion tracking [117], but, in applications for image enhancement, are considered undesired visual artifacts [118]. The phase-aware video denoising method, SMD (introduced in Chapter 1.1) uses supervised manifold learning to denoise biomedical video. The method assumes that images with similar phase should be similar. However, this assumption does not hold in the presence of uncorrelated motion, such as global motion caused by sensor motion. This experiment evaluates the effect of using our proposed algorithm as a preprocessing step to phase-aware video denoising.

For quantitative evaluation, we applied the SMD algorithm to the same synthetic data set introduced in Chapter 3.3.3 and used two metrics, Mean Structural Similarity (MSSIM) and Ultrasound Despeckling Assessment Index (USDSAI), to evaluate the denoising performance before and after applying video alignment.

SSIM [104] is a window-based measure, which has been used to evaluate the denoising quality by incorporating three factors for image comparison: luminance, contrast, and structure. Given image $\mathbf{x}_i$ and the denoised image $\hat{\mathbf{x}}_i$, the overall similarity between $\mathbf{x}_i$ and $\hat{\mathbf{x}}_i$ can be computed as the mean of SSIM values (MSSIM) at each pixel location. MSSIM ranges from 0 to 1, and larger values indicate better denoising performance. For these tests, the window size is 20×20, and the predefined constants used to prevent numerical instability, c_1 and c_2, are 0.01 and 0.03, respectively. For the images generated using the Field II simulator, we use the binary ground truth masks

Table 3.1: Quantitative results for denoising synthetic data using SMD. For both metrics, higher is better.

	MSSIM	USDSAI
SMD	0.784 ± 0.040	2.706 ± 1.275
`Healy2007` + SMD	0.780 ± 0.056	2.951 ± 1.750
`Wu2014` + SMD	0.807 ± 0.047	3.853 ± 1.884
`VSPAK` + SMD	**0.826 ± 0.030**	**4.169 ± 1.399**

to indicate the foreground-background segmentation and take the average intensity of the noisy images in each region to serve as the ground truth intensity values.

USDSAI [89] is a denoising metric designed for images with distinct foreground, $\mathcal{S}_\mathrm{f}$, and background, $\mathcal{S}_\mathrm{b}$, intensity classes. After denoising, pixel intensities should show high within-class agreement and low inter-class agreement. Given denoised images, Q_alg measures the degree of discrimination between the two classes:

$$Q_\mathrm{alg} = \frac{(\mu_\mathrm{f} - \mu_\mathrm{b})^2}{\sigma_\mathrm{f}^2 + \sigma_\mathrm{b}^2}, \tag{3.8}$$

where μ_f and μ_b are the mean of the two classes, and σ_f^2 and σ_b^2 are the standard deviations. Q_0 is the value of this measure for the (noisy) input image. USDSAI is defined as the ratio, $\frac{Q_\mathrm{alg}}{Q_0}$, and larger values indicate better noise reduction performance.

Table 3.1 shows quantitative results for denoising the synthetic data. Preprocessing the data using the single reference method, `Healy2007`, shows no improvement over using the SMD algorithm with unprocessed images. Both of the methods based on multiple keyframe images show significant improvement compared to applying SMD to unaligned data ($p < 0.001$). Additionally, `VSPAK` shows a significant improvement compared to the recent multi-reference method `Wu2014` on both metrics ($p < 0.001$). Figure 3.10 shows three example frames of the synthetic video denoised using all

the methods. [1] The visual results correspond with the quantitative measures. Applying the SMD method without any alignment results in blurred boundaries on all three frames. These boundaries are more distinct using both `Wu2014` and `VSPAK` as preprocessing steps, with those from `VSPAK` + SMD being sharper.

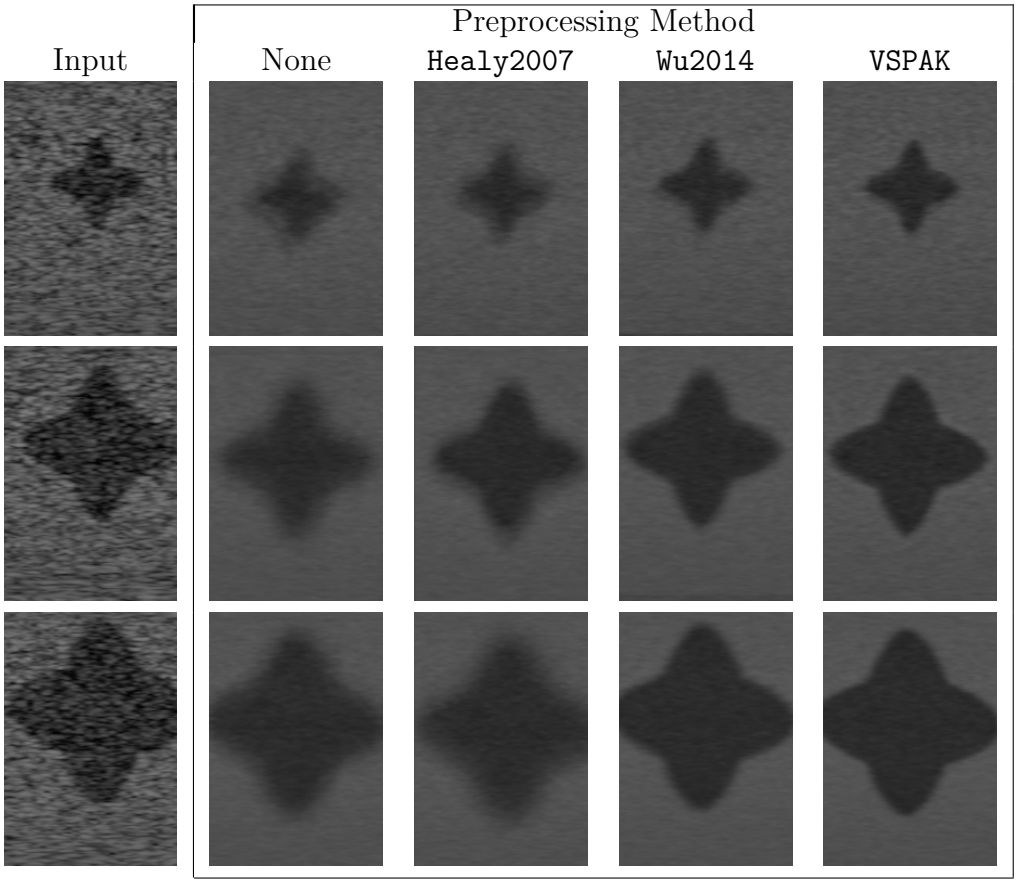


Figure 3.10: Denoising results using the SMD algorithm on synthetic data. Each row shows a (noisy) input frame and the denoised versions with each algorithm used as preprocessing alignment.

Figure 3.11 shows the denoising results, with dotted rectangles highlighting areas with imaging artifacts (e.g., blurred areas, corrupted boundaries). `Wu2014` and `VSPAK` show pronounced improvement over SMD without preprocessing and `Healy2007`, both

[1] Due to video stabilization, some output images appear shifted relative to the input. For each method the ground truth masks were transformed using the estimated alignment parameters for each frame.

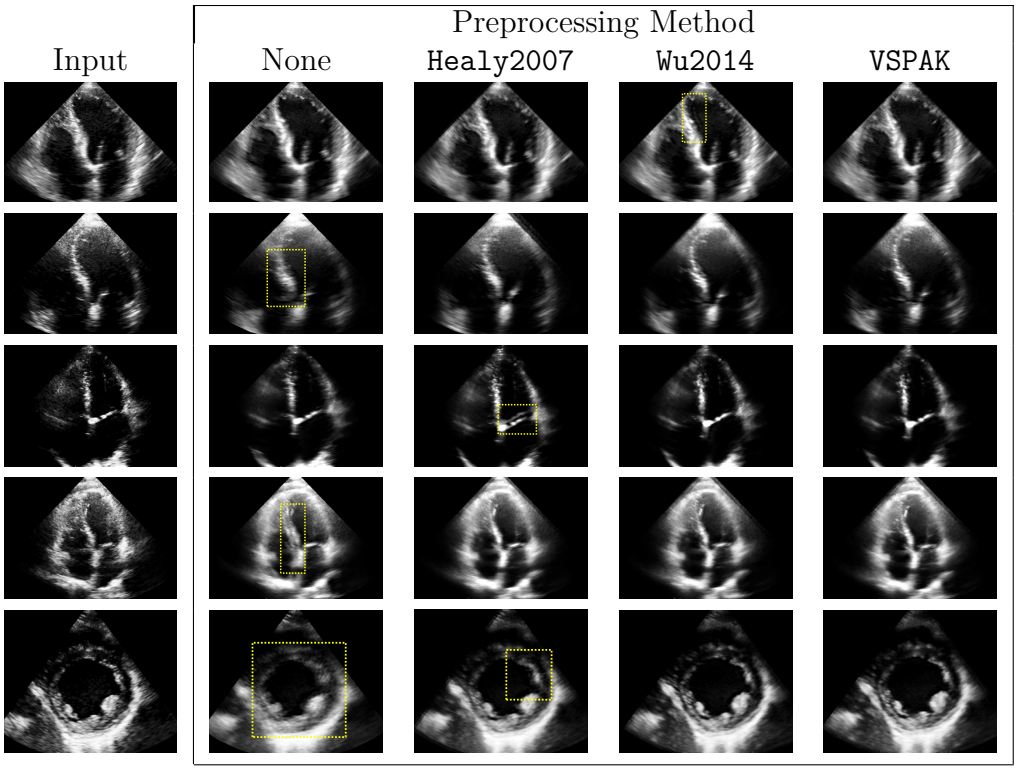


Figure 3.11: Example denoised frames from echocardiogram videos. Each row shows a (noisy) input frame and the denoised versions with each algorithm used as preprocessing alignment. The dotted rectangles highlight areas with imaging artifacts (e.g., blurred areas, corrupted boundaries).

of which produce artifacts such as ghosting (e.g., fourth image of "None", third image of `Healy2007`) and blurring (e.g., second image and fifth images of "None"). While `Wu2014` and `VSPAK` have comparable performance on all images in terms of reducing noise level and preserving structures, `Wu2014` underperforms on the first image as seen from the ghosting effect around the ventricle wall. Overall, the qualitative observations from echocardiogram denoising follow the quantitative and qualitative trends from the synthetic data experiments: `VSPAK` can significantly improve existing medical image analysis algorithms when used as a preprocessing step.

3.4 Summary

We present a keyframe based approach that formulates the image relationships as a weighted graph using the similarities of the variation of interest and factorizes image change by removing auxiliary variation between images with a strong graph connection. The approach is applied to echocardiogram video stabilization with image labels automatically obtained from ECG signals with no manual labeling effort required. We evaluated our approach both quantitatively and qualitatively on multiple data sets and demonstrated its benefit as a preprocessing step for two common echocardiogram applications. The weakly supervised manifold factorization problem we studied in this chapter corresponds with the scenario of weakly supervised image manifold learning where the provided image labels only explain a part of the latent factors of image variation.

CHAPTER 4: ROBUST MANIFOLD REGRESSION

Robust manifold regression considers a variant of weakly supervised image manifold learning, where full image labels are provided, but they are heavily corrupted. Assuming that the labels are noisy samples from a smooth function defined on the manifold, the goal is to incorporate the relationships of images to learn the function. As shown in Figure 4.1, a corrupted label usually does not conform with other labels in a local neighborhood of the manifold and is smoothed after performing robust manifold regression.

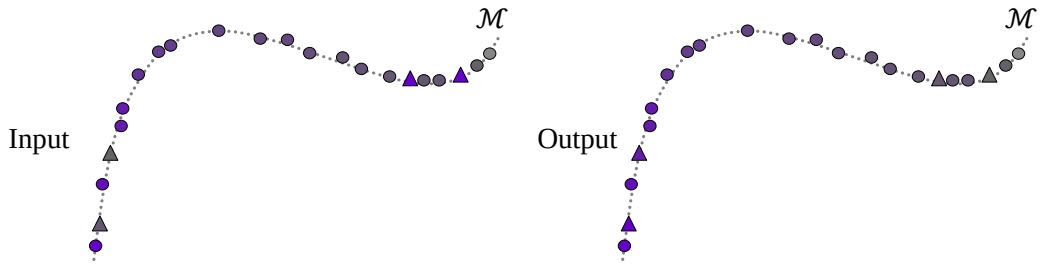
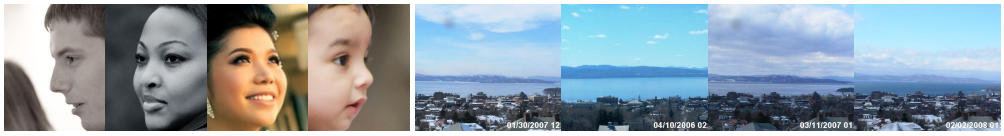


Figure 4.1: Illustration of robust manifold regression. Color indicates label values, and triangles indicate data points originally associated with corrupted label values.

Robust manifold regression is most applicable for cases where image labels are of low accuracy, such as the image labels obtained from certain automated algorithms. A recent trend to acquire image labels is via crowdsourcing or co-located sensors, which effectively automates the label collection process, allowing for the rapid creation of labeled data sets at scales previously impossible. However, label accuracy often suffers.

For example, Figure 4.2 shows representative images from two publicly-available image sets (AMOS [49] and Geofaces [48]) and the associated labels, including instances of mislabeled images. These image labels are heavily corrupted and can not be directly used for supervised learning. However, the labels usually describe some visual concept present in the images and can provide weak supervision.



Original	-90	0	45	45	Clear	Clear	Cloudy	Cloudy
Output	-90	-52	43	90	Partly	Clear	Cloudy	Partly

Figure 4.2: Our method can be applied to image sets with ordered labels (left: head pose estimates, right: cloudiness estimates). For each image, we show the original label (top row) and predicted values from our method (bottom row). The examples in red highlight errors in the original labels.

In this chapter, we present a method to address the problem of robust regression on image manifolds. We take advantage of the fact that these data sets contain semantically-related images whose relationship can be exploited to learn a smooth function of the labels with respect to the images. Unlike traditional robust regression methods, our method utilizes the relationship between the underlying image manifold structure and the visual concepts described by the image labels. We further combine this manifold assumption with sparse regularization, which allows our method to learn the underlying dependency between images and labels even at a very high rate of label corruption.

4.1 Background

There has been much work that involves learning with noisy categorical labels (e.g., [33, 88]), and the general problem of robust classification with mislabeled examples (e.g., [69]). Our work, to our knowledge, is the first to consider this problem in the context of regression, with ordinal or real-valued labels. While most regression techniques are somewhat tolerant to noise, they are generally not designed to handle large amounts of corruption found in the labels from real-world image sets.

The literature on robust regression is vast, spanning approaches from M-estimation to more recent methods designed to overcome the limitations of the commonly-used least squares error measure (e.g., sensitivity to noise and outliers). Robust substitutes have been investigated, including least median of squares [79] and least trimmed squares [3]. Least absolute deviation [100] has seen increased interest with the growing prominence of sparse representations and compressed sensing theory, with applications to computer vision and imaging problems, such as face recognition [109]. Our method also incorporates sparsity as a means of discriminating between noisy and noise-free labels, but additionally correlates the labels to the underlying manifold structure commonly exhibited by natural image sets.

An important family of robust regression methods are Random Sample Consensus (RANSAC) and its variants [34, 77]. They have been successfully applied to a variety of geometric vision problems, such as 3D reconstruction from noisy feature matches [2, 83]. Most RANSAC methods are superlinear (and often exponential) in the number of iterations as a function of the number of model parameters. For geo-

metric vision problems, the number of model parameters is usually small (e.g., 7 for the fundamental matrix). However, for our problem, the model parameters are derived from a high-dimensional image space, and the relationship between the domain and range is unknown and, in most cases, nonlinear. In comparison, our algorithm is non-parametric, data-driven, and the time complexity is not a function of the ambient space dimension.

4.2 Framework

The input for this problem is the image set, $\mathbf{X}$, and associated (noisy) labels, $\mathbf{Y}$. For clarity, in this chapter, the labels are treated as one-dimensional real values; so $\mathbf{Y} = [y_1, y_2, \cdots, y_{\mathrm{N}}]^\intercal$, where $y_i \in \mathcal{R}$. in Chapter 4.4, we present extensions for ordinal and multi-dimensional labels. We assume there exists a smooth function, $f : \mathcal{R}^{\mathrm{D}} \rightarrow \mathcal{R}$, which maps the input image features to the labels, and that the ideal labels, $y_i^\star = f(\mathbf{x}_i)$, are samples from this output manifold. Our goal is to recover the ideal function values, $\hat{\mathbf{Y}} = [\hat{y}_1, \hat{y}_2, \cdots, \hat{y}_{\mathrm{N}}]^\intercal$, such that $\hat{y}_i = y_i^\star$.

The desired property for the manifold function, f, is that, it should change smoothly on the image manifold. However, learning an unknown function on a manifold only defined by images is an ill-posed problem. We propose to use the regularized empirical risk minimization framework:

$$\underset{f}{\operatorname{argmin}} \quad R(f; \mathbf{X}) + \lambda_{\mathrm{r}} L(f; \mathbf{Y}) \tag{4.1}$$

where R regularizes the function on the manifold defined by the images, $\mathbf{X}$, L is a loss function based on the provided image labels, $\mathbf{Y}$, and λ_{r} is the trade-off parameter.

4.3 Method

Many choices are possible for the two terms in Equation 4.1. In this chapter, we introduce in detail our choice for the manifold regularization term, $R(\cdot)$, and the loss function, $L(\cdot)$, motivated by the problem of noisy ordered labels for natural image collections.

4.3.1 Manifold Regularization

Many approaches to manifold regularization have been proposed, which extend some notion of local linearity to a global model of the manifold. One such approach is based on the Hessian regularizer, which has been applied to, for example, nonlinear dimensionality reduction [27] and semi-supervised regression [27].

For a point on the manifold, the local Hessian functional is defined on its associated tangent space as the Frobenius norm of the Hessian matrix. This provides a coordinate system that is isometric to the manifold intrinsic coordinate. The local measure is then averaged over the entire manifold to provide a global measurement, which is an extension of the average Frobenius norm of the Hessian of a function in Euclidean space to manifolds. Minimizing this term leads to locally linear functions. Several properties of the Hessian functional make it useful in our case: (1) it provides a data-driven way for manifold function regularization that enables non-parametric regression; (2) it can handle extrapolation better than other proposed manifold regularizers (e.g., Laplacian) [43]. However, unlike [27], our goal is not to explicitly learn a low-dimensional parametrization of the manifold, but to estimate a function of the labels over the images sampled from the manifold.

For an input point, $\mathbf{x}_i$, let $\mathcal{N}^i$ represent the neighborhood of K nearest neighbors, and $\mathbf{z}_j^{(i)}$ represent the coordinates of $\mathbf{x}_j \in \mathcal{N}^i$ in the d-dimensional tangent space of $\mathbf{x}_i$, where $\mathbf{z}_i^{(i)}$ is defined as the origin. The local Hessian functional estimates a second-order polynomial, f, near $\mathbf{x}_i$ of the form:

$$f = \hat{y}_i + \mathbf{J}_i \mathbf{z}^{(i)} + \frac{1}{2} \mathbf{z}^{(i)\intercal} \mathbf{W}_i \mathbf{z}^{(i)} \tag{4.2}$$

where $\mathbf{J}_i$ and $\mathbf{W}_i$ are the local Jacobian and Hessian matrices, respectively, $\hat{y}_i$ is the predicted label, and $\mathbf{z}^{(i)}$ is the d-dimensional tangent space coordinate. Equation 4.2 is linear with respect to $\mathbf{J}_i$ and $\mathbf{W}_i$. Let $\mathbf{\Psi}_i$ denote the design matrix on neighborhood, $\mathcal{N}^i$, and each row of $\mathbf{\Psi}_i$ corresponds to a neighboring point, $\mathbf{x}_j$:

$$[\mathbf{z}_{j1}, \cdots, \mathbf{z}_{jd}, \mathbf{z}_{j1}\mathbf{z}_{j1}, \mathbf{z}_{j1}\mathbf{z}_{j2}, \cdots, \mathbf{z}_{jd}\mathbf{z}_{jd}] \tag{4.3}$$

where $\mathbf{z}_{jd}$ (superscript omitted for clarity) represents the d-th dimension of $\mathbf{z}_j^{(i)}$. Substituting the predicted values of the labels at the local neighborhood, denoted by $\hat{\mathbf{Y}}^{(i)}$, for the unknown function, f, the least-squares solution for the parameters of the local Jacobian and Hessian matrices is given by:

$$\mathbf{\Psi}_i \begin{bmatrix} | \\ \mathbf{J}_i \\ | \\ \breve{\mathbf{W}}_i \\ | \end{bmatrix} = \hat{\mathbf{Y}}^{(i)} - \hat{y}_i \cdot \mathbf{1} \tag{4.4}$$

where $\breve{\mathbf{W}}_i$ represents the upper triangular portion of $\mathbf{W}_i$, $\mathbf{1}$ is a K-length column

vector of 1, and $\mathbf{J}_i$ and $\breve{\mathbf{W}}_i$ are converted to column vectors. Multiplying both sides of Equation 4.4 by the pseudo-inverse of the design matrix and taking only the bottom $d + d(d+1)/2$ rows of both sides, we get:

$$\breve{\mathbf{W}}_i = \mathbf{\Psi}^\dagger \left(\hat{\mathbf{Y}}^{(i)} - \hat{y}_i \cdot \mathbf{1} \right) \tag{4.5}$$

where $\mathbf{\Psi}^\dagger$ represents the bottom $d + d(d+1)/2$ rows of the pseudo-inverse of the design matrix. Including contributions from the $\hat{y}_i$ term in Equation 4.4, the right side can be written as $\tilde{\mathbf{\Psi}}^\dagger \hat{\mathbf{Y}}^{(i)}$. Scaling the rows in $\tilde{\mathbf{\Psi}}^\dagger$ corresponding to the diagonal elements of $\mathbf{W}_i$ by 2 and those corresponding to off-diagonal elements by $\sqrt{2}$. we get the following expression for the approximation of local Hessian functional:

$$\begin{aligned} \|\mathbf{W}_i\|_F^2 &= \sum_r \left(\tilde{\boldsymbol{\psi}}_r^\dagger \hat{\mathbf{Y}}^{(i)} \right)^2 \\ &= \left(\hat{\mathbf{Y}}^{(i)} \right)^\intercal \mathbf{H}_i \hat{\mathbf{Y}}^{(i)} \end{aligned} \tag{4.6}$$

where

$$\mathbf{H}_i = \sum_r \left(\tilde{\boldsymbol{\psi}}_r^\dagger \right)^\intercal \left(\tilde{\boldsymbol{\psi}}_r^\dagger \right) \tag{4.7}$$

and $\tilde{\boldsymbol{\psi}}_r^\dagger$ denotes the r-th row of $\tilde{\mathbf{\Psi}}^\dagger$. The global Hessian estimator is the sum of the local estimators over all the input points. Let $\tilde{\mathbf{H}}_i$ denote the sparse N $\times$ N version of $\mathbf{H}_i$ where $\tilde{\mathbf{H}}_i$ and $\mathbf{H}_i$ are identical at the locations corresponding to points in $\mathcal{N}^i$ and 0 otherwise. So,

$$\mathbf{H} = \sum_{i=1}^{\mathrm{N}} \tilde{\mathbf{H}}_i \tag{4.8}$$

and the global regularizer of the manifold function can be obtained in the quadratic form, $\hat{\mathbf{Y}}^\intercal \mathbf{H} \hat{\mathbf{Y}}$.

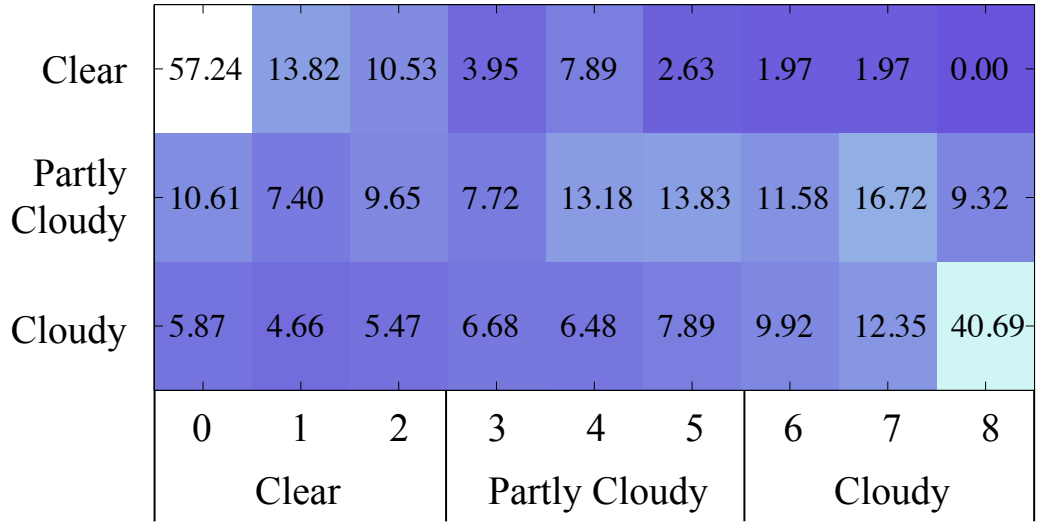


Figure 4.3: Distribution of the mislabeled examples. For a data set of outdoor images with cloudiness metadata (measured in okta from 0-8), the confusion matrix shows the distribution of the input label (columns) with manual annotations (rows).

4.3.2 Loss Function

Modeling the noise of labels associated with large image collections can be difficult. Labels can be obtained from automated algorithms, co-located sensors, or crowdsourcing; each of which introduces different types of error. In our work, we observed that much of this data was corrupted nearly uniformly and not necessarily biased toward the ground truth. For example, consider the AMOS data set [49], which provides weather metadata associated with images captured from globally-distributed webcams. One label is cloud okta, a cloudiness measure that ranges from clear (0) to cloudy (8). Figure 4.3 shows a confusion matrix of the cloudiness values between the AMOS labels and manual annotations for a representative subset of 1000 images. This pattern of roughly uniformly distributed noise is consistent with research into labels obtained via crowdsourcing (e.g. Amazon Mechanical Turk) where "bad" users tend to provide information uncorrelated with the correct answer [78].

This suggests that the commonly-used $L2$ error measure is not well-suited to the problem, as it often results in poor performance for non-normal noise distributions [100]. The $L1$ norm, however, is robust to high variance in noise and implicitly promotes sparsity in the residual error. This is the desired behavior, since sparsity in the residual error allows for soft subset selection of "good" labels and de-emphasizes the contribution of labels with extreme noise. In Chapter 4.4, we compare the performance of our method using both $L1$ and $L2$ loss.

4.3.3 Optimization

Combining the Hessian regularization term with the $L1$ loss, we are left with the following optimization:

$$\underset{\hat{\mathbf{Y}}}{\operatorname{argmin}} \quad \hat{\mathbf{Y}}^{\intercal}\mathbf{H}\hat{\mathbf{Y}} + \lambda_{\mathrm{r}}\|\hat{\mathbf{Y}} - \mathbf{Y}\|_1 \tag{4.9}$$

In order to efficiently solve Equation (4.9) for the denoised labels, $\hat{\mathbf{Y}}$, we show that the global Hessian estimator, $\mathbf{H}$, is positive semidefinite (PSD).

Proof. First, the local Hessian estimator, $\mathbf{H}_i$, is PSD. In Equation 4.7, each term in the summation can be represented as the product of a matrix and its transpose, which is PSD. Next, we show that the sparse variant of the local estimator, $\tilde{\mathbf{H}}_i$, is PSD. Let $\boldsymbol{\nu}$ be a column vector of length N, so

$$\boldsymbol{\nu}^{\intercal}\tilde{\mathbf{H}}_i\boldsymbol{\nu} = \sum_{j=1}^{\mathrm{N}}\sum_{l=1}^{\mathrm{N}} \tilde{h}_{jl}\nu_j\nu_l$$

where $\tilde{h}_{jl}$ is the entry of $\tilde{\mathbf{H}}_i$ at the j-th row and l-th column. Since $\tilde{\mathbf{H}}_i$ is a sparse matrix that contains the same elements of $\mathbf{H}_i$ at the intersections of rows and columns

corresponding to $\mathcal{N}^i$, the above equation is reduced to a sum of K^2 terms:

$$\begin{aligned}\boldsymbol{\nu}^\intercal \tilde{\mathbf{H}}_i \boldsymbol{\nu} &= \sum_{j \in \mathcal{N}^i} \sum_{l \in \mathcal{N}^i} \tilde{h}_{jl} \nu_j \nu_l \\ &= \hat{\boldsymbol{\nu}}^\intercal \mathbf{H}_i \hat{\boldsymbol{\nu}} \geq 0\end{aligned}$$

where $\hat{\boldsymbol{\nu}}$ is a K-length column vector of elements from $\boldsymbol{\nu}$ at positions $\mathcal{N}^i$. Therefore, $\tilde{\mathbf{H}}_i$ is PSD. Finally, we get the global Hessian estimator, $\mathbf{H}$, is PSD since it is the sum of the N sparse local PSD matrices, $\{\tilde{\mathbf{H}}_i\}_{i=1}^{\mathrm{N}}$. □

Therefore, Equation (4.9) is a convex quadratic program with $L1$ regularization. Performing Cholesky decomposition on $\mathbf{H}$, we get $\mathbf{H} = \Delta^\intercal \Delta$ and are left with:

$$\underset{\hat{\mathbf{Y}}}{\operatorname{argmin}} \quad \|\Delta \hat{\mathbf{Y}}\|_2^2 + \lambda_{\mathrm{r}} \|\hat{\mathbf{Y}} - \mathbf{Y}\|_1 \tag{4.10}$$

where Δ is a sparse N × N upper triangular matrix. This convex optimization can be solved using standard algorithms, or using more efficient solvers specialized for large-scale, sparse $L1$-regularized least squares problems [54].

4.3.4 Algorithm

Given a set of images and (noisy) labels, our method, *Hessian-Regularized Robust Regression* (`H3R`), outlined in Algorithm 2, returns the denoised labels.

For our method, the intrinsic dimension of the method, d, and the neighborhood size, K, can be provided using prior knowledge or estimated directly from the data. In Chapter 4.4, we describe the implementation details for `H3R`.

Algorithm 2 `H3R`

Input: images, $\mathbf{X}$; labels, $\mathbf{Y}$;
Output: estimated labels, $\hat{\mathbf{Y}}$

1: Estimate subspace dimension, d, and neighborhood size, K
2: **for all** $\mathbf{x}_i \in \mathbf{X}$ **do**
3: Find $\mathcal{N}^i$, the K-nearest neighbors of $\mathbf{x}_i$
4: Perform PCA on neighborhood, $\mathcal{N}^i$, to obtain d-dimensional tangent space coordinates
5: Construct design matrix, $\mathbf{\Psi}_i$ (Eq. 4.3)
6: Compute local Hessian estimator (Eq. 4.7)
7: Construct global Hessian estimator, $\mathbf{H}$ (Eq. 4.8)
8: Solve for $\hat{\mathbf{Y}}$ (Eq. 4.10)

4.4 Experimental Evaluation

We evaluate the performance of `H3R` on a diverse set of labeled image collections and compare the results against the following regression methods.

- K-`NN`: The label of each point is estimated as the average labels of its K nearest neighbors in the data set, where K is set to the same value used by our method.

- Radial basis function network (`RBFN`) [72]: The neural network contains $\sqrt{\mathrm{N}}$ hidden layer nodes with kernel width equal to the average distance to the 2-nearest cluster centers.

- `RANSAC` [77]:[1] The threshold for inliers is set to the 10% of the label dynamic range, and maximum number of iterations is set to 10^7.

- ϵ support vector regression (`SVR`) [17] with the radial basis kernel. The kernel width is set to the average Euclidean distances of the input, and the inlier

[1] The linear model of `RANSAC` learns $\mathrm{D}+1$ parameters, where D is the dimensionality of the input. To make the problem tractable, for image data, we applied PCA to preserve 80% of the variation, which resulted in an input dimensionality of $\sim$20 across the data sets. Higher-order models were computationally prohibitive.

threshold, ϵ, is set to the 10% of the label dynamic range.

- Kernel Supervised PCA (`KSPCA`) [7]: for both the input data and labels, the radial basis kernel is used with the kernel width set to the average Euclidean distance.

We used labeled data sets with known ground truth. For each data set, the labels are normalized to $[0, 1]$.

- *Swiss Roll*, commonly used to evaluate machine learning algorithms, consists of 5000 points randomly sampled from a 2D manifold embedded in 3D. For each 3D example, the real-valued label is defined as sine of the geodesic distance to the center point on the manifold.

- *Paper Boy Statue* [75] consists of 840 images of a rigid object on a turntable platform captured from a camera on an elevating arm. The images are captured every 6 degrees of rotation from 0–354 and every 6 degrees of elevation from 6–84. Each image is cropped and subsampled to 32×20, represented as a pixel intensity vector, and noise was added to the elevation and rotation angles. To account for the cyclic rotation parameter, we take the values as cylindrical coordinates (with unit radius) and convert to a 3D Euclidean parametrization. Results are reported as rotation and elevation angles.

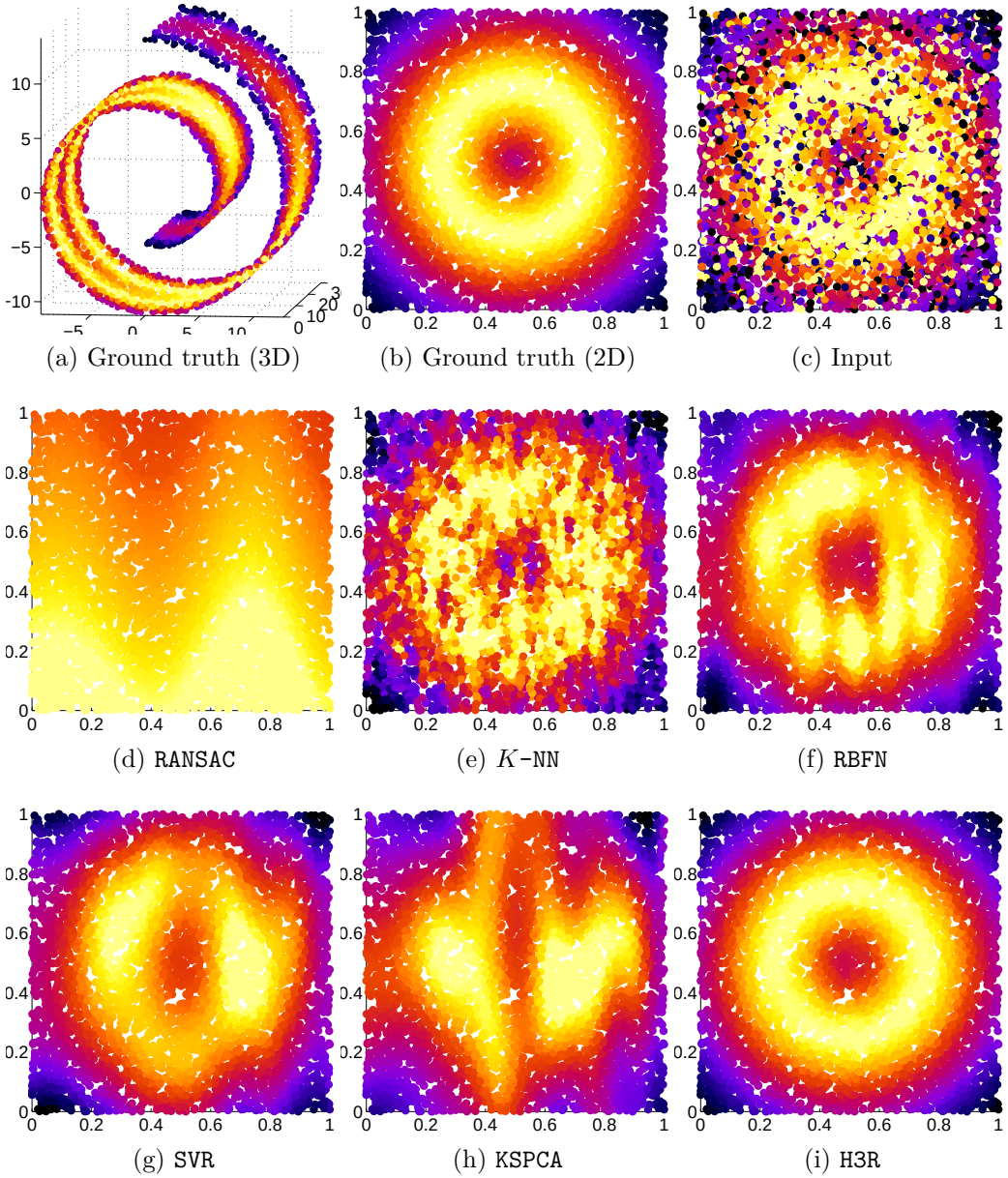
- *Digit* [53] consists of 10,000 images of the digit "1", with four degrees of variations: horizontal translation, vertical translation, rotation and thickness. Each image is represented as a vector of raw pixel values.

Table 4.1: RMSE of H3R on the Swiss Roll data using $L1$ or $L2$ loss. Noise was generated using the Laplacian (b = 0.05), Gaussian ($\sigma = 0.5$), uniform additive ($[-1, 1]$, 50% corruption), and salt & pepper (50%) noise models.

	Lap.	Gauss.	Unif.	S&P
$L1$	**0.067**	**0.090**	**0.013**	**0.014**
$L2$	0.068	0.136	0.112	0.197

For H3R, the intrinsic manifold dimensionality, d, neighborhood size, K, and trade-off parameter, λ can be provided if prior knowledge is available. However, these values can be directly estimated from the data, leaving no free parameters to the system. To estimate the intrinsic manifold dimensionality, d, we apply local PCA on a neighborhood of 20 points from a small set of randomly selected examples and set d as the value corresponding to the 'elbow point' of the residual variance curve. The number of nearest neighbors, K, is loosely related to the manifold intrinsic dimension. We found that the method was robust to the value of K, and empirically determined that $K = 5d$. For the regularization parameter, λ_r, we use the L-curve method [39] to select a value in the range $[10^{-10}, 10^5]$. The algorithm is implemented in Matlab, and we use the *l1-ls* package [54] for $L1$-regularized least squares optimization. The computation of the algorithm is dominated by the optimization step. On a standard PC, with an input of 1,000 samples, our method takes less than 5 seconds, on average.

To evaluate the choice of loss function in our method, we performed manifold regression using the Swiss Roll data set (Figure 4.4) corrupted by commonly-used artificial noise models. Table 4.1 shows the root-mean-square error (RMSE) values of the predicted output from H3R. The order of noise models (left to right) represents moderate to high noise levels, and across all of the settings, the $L1$ norm outperforms



(a) Ground truth (3D) (b) Ground truth (2D) (c) Input

(d) RANSAC (e) K-NN (f) RBFN

(g) SVR (h) KSPCA (i) H3R

Figure 4.4: For the Swiss Roll (50% label corruption), the color in each plot indicates the manifold function value. For clarity, (b) to (i) are plotted using 2D manifold coordinates.

the choice of $L2$, often by a wide margin. For the remaining experiments, we use the $L1$ loss with `H3R`.

4.4.1 Robust Regression

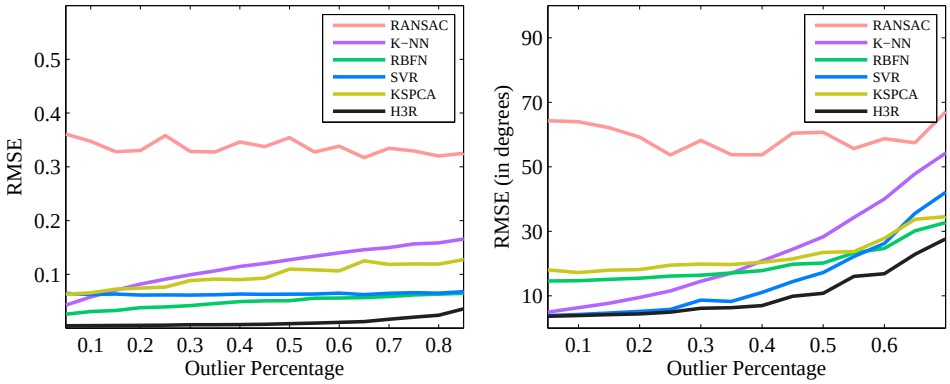
For randomly-selected subsets of examples of varying size, the labels are corrupted by adding uniform noise in the range $[-1, 1]$. Each method is provided the (corrupted) labeled data as input. For multi-dimensional labels, each label is predicted independently for consistency across the methods. Figure 4.5 shows the results of these experiments, reported as the average RMSE in the predicted values from 10 repeated trials. Across all of the experiments, `H3R` returns the closest predicted values, even at corruption rates as high as 80%. In all but two cases, `RANSAC` performed poorly. This is expected as `RANSAC` requires a pre-specified model, and the linear model is not a reasonable choice for these experiments, as the relationship between the input and labels is nonlinear in most cases. While the remaining methods should be better-suited to nonlinear regression, for the problem of estimating the elevation of the camera for the Paper Boy Statue, `RANSAC` returned the next best predictions. In aggregate, `SVR`, `KSPCA`, K-`NN`, and `RBFN` showed similar performance with little consistency in relative performance across the experiments.

The Swiss Roll data allows for both the manifold and function defined on the manifold to be easily visualized. Figure 4.4 shows the ground truth, corrupted input, and regression results from each method for an trial with a 50% corruption rate. This is a 3D problem (Figure 4.4(a)); however, for clarity, the graphs in Figure 4.4 are plotted using 2D manifold coordinates. K-`NN` locally smooths the label noise,

but the global model remains discontinuous and noisy. RBFN, SVR, and KSPCA all learn smooth functions on the manifold, however, the recovered function deviates substantially from ground truth. The result from H3R closely matches the ground truth (RMSE ≈ 0), and remains nearly perfect up to a corruption rate of 60%.

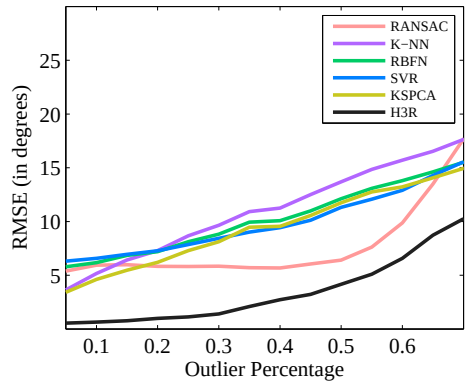
Figure 4.6 shows the output from each method at 50% label corruption for the Paper Boy Statue data set. Compared to H3R, the other methods include substantially more misplaced images, which indicate incorrect predictions for rotation, elevation, or both. While H3R returned the best predictions for both rotation and elevation, there were differences in the patterns of results. The change in elevation appears to be approximately linear, as RANSAC outperformed the nonlinear approaches (except for H3R) and was able to achieve low error rates (RMSE $\approx 5^\circ$) on up to 50% corrupted labels. This was not the case for the nonlinear transformation represented by turntable rotation, where RANSAC was the worst performer. However, for these different transformations, our method learned different accurate, smooth functions on the same image manifold.

Similar results are observed with the Digit data. Figure 4.7 shows results for an experiment with 50% label corruption. Each group shows the images sorted by the listed parameter, with the remainder fixed. So, in the ideal case, there should only be a single smoothly varying transformation (e.g., rotation) across each row. Non-smooth changes from left to right or auxiliary changes from other transformations indicate an inaccurate prediction. The visual results align with the quantitative results. This is a challenging 4-dimensional prediction; H3R is the top-performer for each of the modes of image variability and returns low errors (RMSE < 0.05) at a corruption rate up

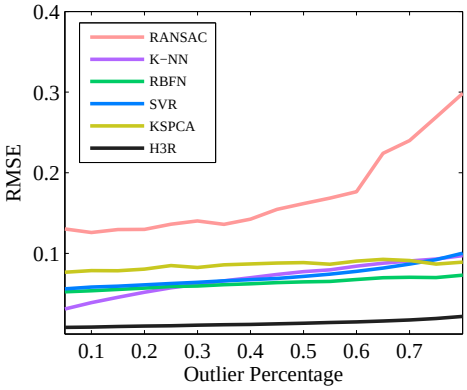


(a) Swiss Roll

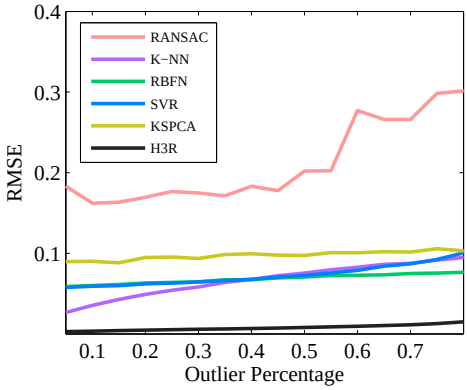
(b) Statue Rotation



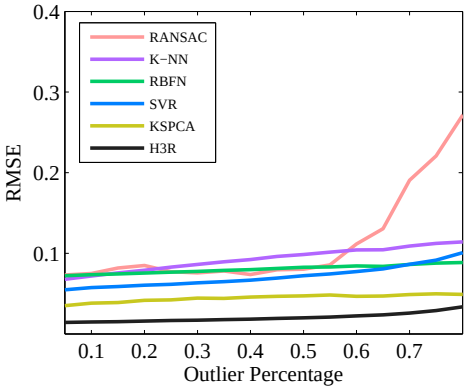
(c) Statue Elevation



(d) Digit Horiz. Trans.

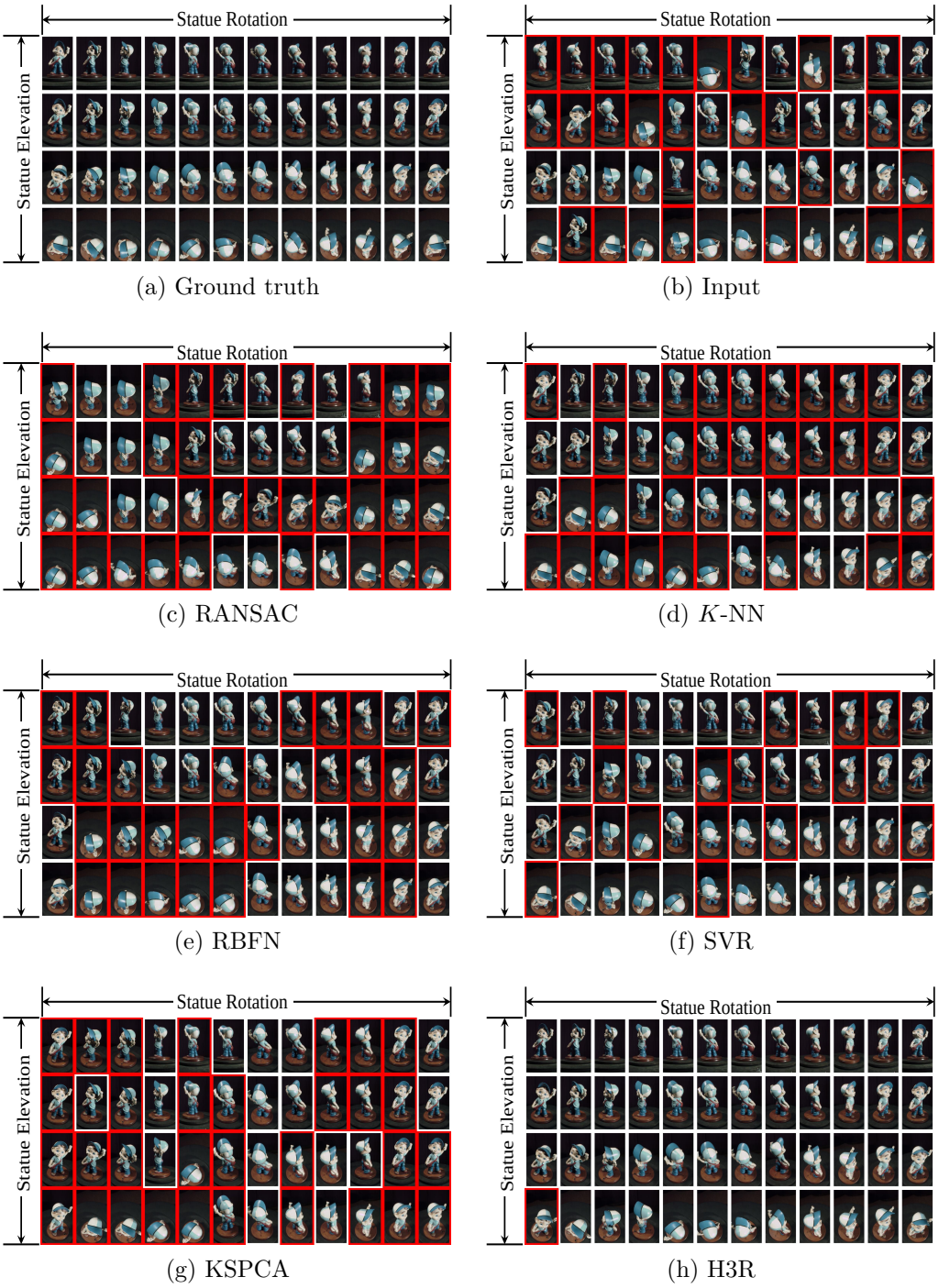


(e) Digit Rotation



(f) Digit Thickness

Figure 4.5: RMSE values for the predicted labels on the Swiss Roll (a), Paper Boy Statue (b,c), and Digit (d-f) data sets with varying label corruption rate. (The results for vertical translation for the Digit data set closely followed that for horizontal translation.)



(a) Ground truth (b) Input

(c) RANSAC (d) K-NN

(e) RBFN (f) SVR

(g) KSPCA (h) H3R

Figure 4.6: For Statue data set with 50% corruption, this figure shows images sampled from an 2D grid of angles in elevation $[6°, 84°]$ and rotation $[0°, 359°]$. Red bounding boxes highlight images with elevation or rotation error $> 10°$.

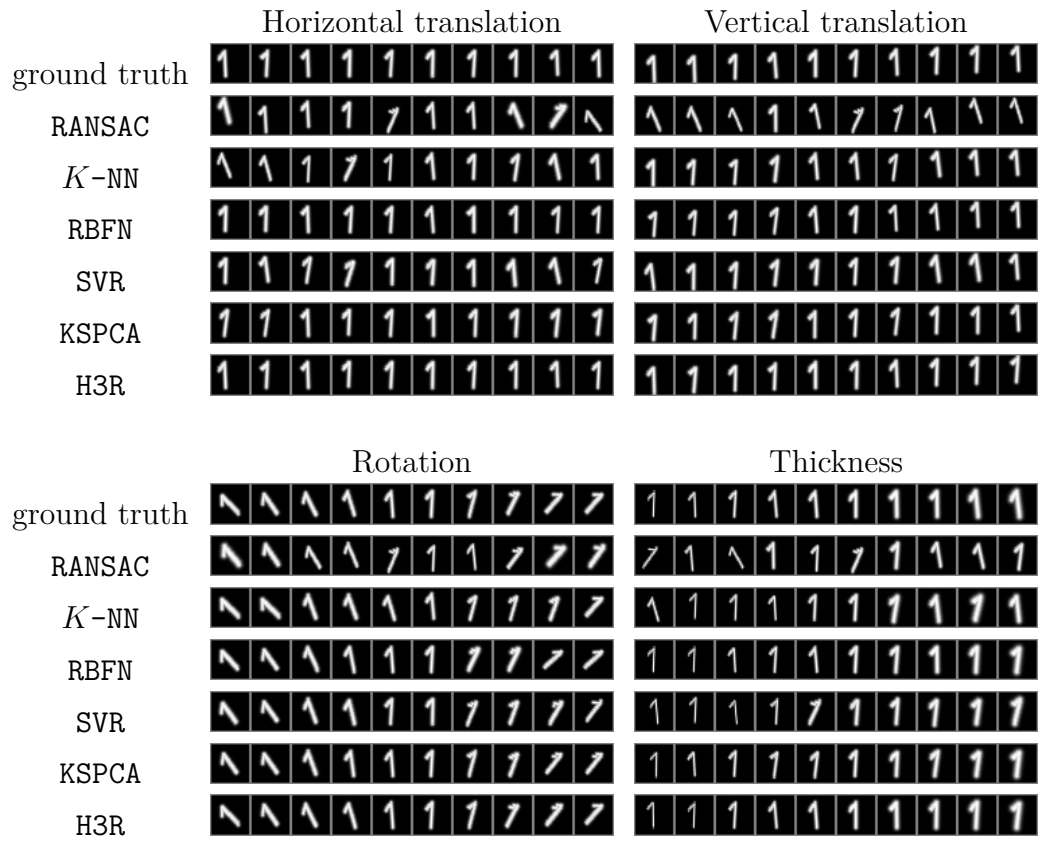


Figure 4.7: Each row shows images sorted by the predicted label. For each group, the specified (normalized) transformation should smoothly vary from 0 to 1, with the other labels fixed to 0.5. Non-smooth changes from left to right or auxiliary changes from other transformations indicate an incorrect prediction.

to 80%. This demonstrates the ability of our method to learn a variety of different functions on image manifolds.

4.5 Applications to Ordered Label Denoising

In this chapter, we apply `H3R` to the problem of denoising ordered labels from real-world, large-scale, publicly available data sets used as computer vision benchmarks. For clarity of presentation, we only include the top three related methods (`RBFN`, `SVR`, and `KSPCA`) for comparison. As opposed to quantitative measures of error, we interpret these results by visual inspection as ground truth is unavailable.

4.5.1 Weather from Images

The Archive of Many Outdoor Scenes (AMOS) [49, 47] is a repository of millions of images captured from globally-distributed webcams. In addition to images, AMOS provides associated weather metadata. While some of these parameters (e.g., air pressure, wind velocity) do not affect the appearance of the images, others, such as measures of cloud cover, can be important for methods in outdoor scene analysis. Some algorithms use clouds as a visual cue, while others assume cloudless imagery. Cloud okta, collected with AMOS images, is a measure of cloudiness from clear (0) to cloudy (8). These weather values are estimated from the closest weather stations, which may be far enough to be under different weather conditions from where the image is captured. This results in inaccurate labels, rendering cloudiness-based filtering unreliable.

Each AMOS image is represented using the 16-dimensional bag-of-colors feature [107], and the images are grouped by the originating webcam. Figure 4.8 shows representative results from various scenes. Those boxed in red are examples where the original label does not appear to match the cloud level depicted in the scene. The 2^{nd} example shows a case where `RBFN`, `SVR`, and `KSPCA` incorrectly changed a seemingly accurate label. The 4^{th} example shows a challenging scene that was both originally mislabeled and not corrected by any of the approaches. Overall, each of the methods improved upon the original labels, with H3R providing predictions that most closely matched the visual appearance of the scene.

RBFN	0.3	5.9	2.0	0.7	2.5	4.6	3.2	7.0	8.0	4.7
SVR	0.8	3.9	2.7	0.6	3.0	3.9	2.2	7.0	7.8	7.0
KSPCA	0.4	5.3	2.3	0.6	2.0	4.3	2.4	6.3	8.0	6.6
H3R	0.8	1.3	2.0	0.8	2.5	3.6	4.7	6.4	8.0	7.3

Figure 4.8: Each image shows the original cloudiness label, which ranges, from 0 (clear) to 8 (cloudy). For each method, the predicted value is shown. Clearly mislabeled (input or predicted) values are indicated by the red text and boxes.

4.5.2 Face Pose Estimation

Many widely used data sets for face analysis, including PubFig [55] and GeoFaces [48], rely on the same algorithm to annotate faces extracted from images collected from the Web or social networking sites. One of the provided parameters is an estimate of the pose of the face as one of five quantized directions: -90, -45, 0, 45, 90. This parameter would be used to, for example, retain only front-facing subjects.

Figure 4.9 shows the results of an experiment with 1,000 randomly selected images from GeoFaces. Each facial image patch is represented using HOG [24] features with a cell size of 50×50 and 9 orientation bins. The first row shows sample faces with the associated pose estimate. Each of the subsequent rows show the same subset of images sorted by the denoised head pose estimate. The red boxes indicate examples where the pose estimate does not visually match the direction the subject is facing. `RBFN`, `SVR`, and `KSPCA` all improved upon the original labels and performed similarly in terms of the number of mislabeled predictions, even though the errors occurred in different regions of the label space. H3R outperformed each of the competing approaches, resulting in no grossly mislabeled examples.

Figure 4.9: For each row, the images are shown with the (input or predicted) head pose estimate. Clearly mislabeled examples are highlighted by red boxes.

4.6 Summary

We presented an algorithm for robust regression on image manifolds and applied it to the problem of ordered label denoising for natural image sets with labels collected from automated algorithms. We demonstrate that by incorporating weak supervision provided by noisy labels with the latent structure of image sets, our non-parametric and computationally efficient approach outperforms related regression methods on a variety of denoising tasks over with 70% label corruption. Although there has been some research that utilizes noisy labels obtained from automated algorithms, the bulk of the algorithms in computer vision still rely on traditional methods for image annotation, which is infeasible and costly on large-scale image sets. Robust manifold regression is a special scenario of weakly supervised image manifold learning where the provided image labels are highly corrupted, which is most applicable for large-scale image sets with labels obtained from automated algorithms.

CHAPTER 5: SEMI-SUPERVISED MULTI-OUTPUT MANIFOLD REGRESSION

In semi-supervised multi-output manifold regression, we consider the case of weakly supervised image manifold learning where image labels are multi-dimensional and are only provided for a subset of the images. The goal is to learn a smooth function that maps from the image space to the multi-dimensional label space regularized by the low-dimensional structure of both the image manifold and the label manifold. As shown in Figure 5.1, the learned labels of originally unlabeled points change smoothly on the image manifold and lie on a low-dimensional structure in the label space.

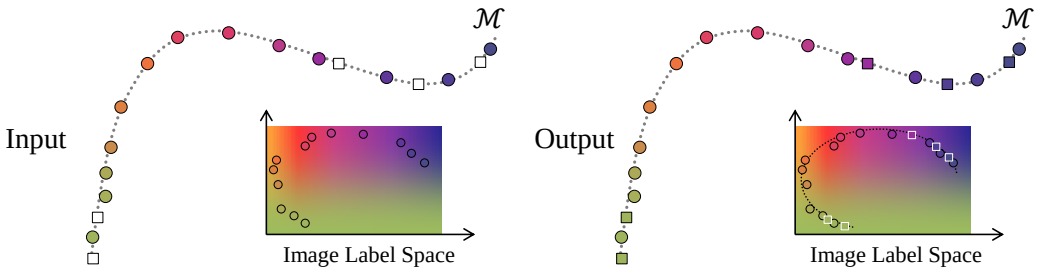


Figure 5.1: Illustration of semi-supervised multi-output manifold regression using a partially labeled toy data set with 2D labels. Square points indicate originally unlabeled data.

Multi-output learning corresponds to many important applications in computer vision, such as contour-based segmentation and articulated pose estimation (Figure 5.2). Compared to categorical or real-valued labels, the issue of acquiring image annotation is exacerbated with multi-dimensional output. Usually, domain expertise

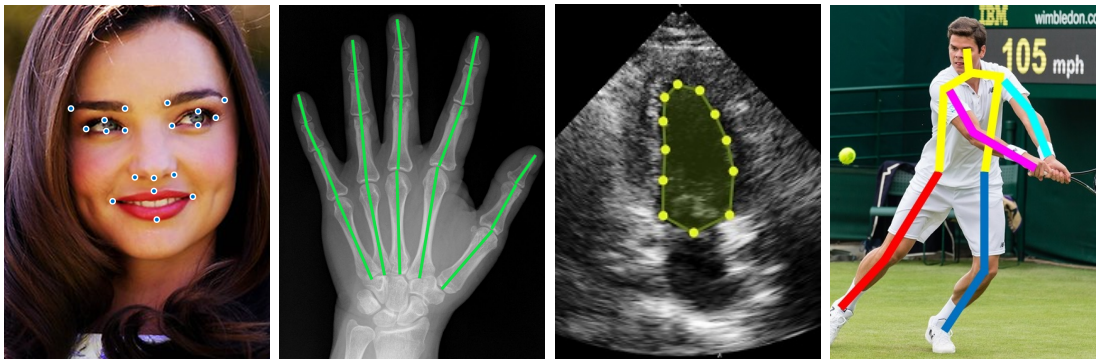


Figure 5.2: Semi-supervised multi-output manifold regression provides a domain-agnostic method for a variety of tasks, including segmentation and pose estimation.

is needed to ensure an accurate annotation, which makes recent approaches for large-scale image metadata acquisition (i.e., crowdsourcing) not always applicable. In this chapter, we aim to balance the ever-increasing availability of visual data with the expense incurred for multi-output labels by investigating the semi-supervised setting of multi-output learning.

We present a semi-supervised multi-output algorithm designed for image manifold regression. Most previous methods for regression focus on regularization in the domain. Our approach considers the manifold structure of both the images and multi-dimensional labels and applies regularization in both spaces. This approach allows our method to learn a semantically meaningful mapping from images to the complex labels, even in the presence of noisy examples.

5.1 Background

Multi-output prediction can be considered a subclass of structured output learning. Beyond real-valued vectors, structured output includes complex label types, such as strings and trees. Many approaches extend traditional supervised learning methods (e.g., SVM [92], boosting [82]) to handle structured data. These approaches have

been applied to, among others, early event detection [45] and human interaction localization in videos [73]. Additionally, there have been extensions to the semi-supervised setting [12, 4, 64]. The main drawback is that most of these approaches require task-specific models for the joint feature space of the input and labels. In some cases, modeling these joint kernels and defining efficient searches in these joint spaces is tantamount to designing a specialized application for the task.

Our method, which leverages the underlying manifold structure of both the image and label spaces to learn smoothly-varying, multi-dimensional labels, uses a data-driven approach and is not sensitive to the change of learning tasks. In fact, we demonstrate the competitive performance of our algorithm compared to task-specific algorithms.

5.2 Framework

For this problem, we are given the images, $\mathbf{X}$, and corresponding multi-dimensional partial labels, $\mathbf{Y} = [\mathbf{y}_1, \mathbf{y}_2, \cdots, \mathbf{y}_{\mathrm{N}}]^\intercal$, where $\mathbf{y}_i \in \{\mathcal{R}^{\mathrm{D}_y}, \varnothing\}$. We assume that, there exists a smooth function, $f : \mathcal{R}^{\mathrm{D}} \rightarrow \mathcal{R}^{\mathrm{D}_y}$, which maps the input image features to the labels, and that the ideal labels, $\mathbf{y}_i^\star = f(\mathbf{x}_i)$, are samples from the output manifold, $\mathcal{M}_{\mathrm{y}}$. Our goal is to predict the label set, $\hat{\mathbf{Y}}$, such that $\hat{\mathbf{y}}_i = \mathbf{y}_i^\star$.

We follow a regularized empirical risk minimization framework to learn the unknown multi-dimensional output function, $f : \mathcal{R}^{\mathrm{D}} \rightarrow \mathcal{R}^{\mathrm{D}_y}$:

$$\underset{f}{\operatorname{argmin}} \quad R(f; \mathbf{X}) + \lambda_{\mathrm{s}} S(f; \mathbf{Y}) + \lambda_{\mathrm{l}} L(f; \mathbf{Y}) \tag{5.1}$$

where R regularizes the function over the input image manifold, S regularizes the

function over the label manifold, L is a loss function (e.g., $L2$ norm) on the subset of labeled examples, and λ_{s} and λ_{l} are the trade-off parameters. In the next chapter, we describe our choices for each of the regularization terms.

5.3 Method

Similar to robust manifold regression, we assume that as the input images vary along on the manifold, the associated labels also change smoothly. We adopt the same manifold regularization term for the semi-supervised multi-output problem. Extending the Hessian regularized term to the case of multi-dimensional output, gives the global regularizer in the quadratic form:

$$\mathrm{Tr}(\hat{\mathbf{Y}}^{\intercal}\mathbf{H}\hat{\mathbf{Y}}) \tag{5.2}$$

Next, we introduce our method for incorporating the output manifold structure.

5.3.1 Label Space Regularization

Similar to the case for the input images, we assume that the multi-dimensional labels only sparsely sample the D_y-dimensional label space and lie on or near a d_y-dimensional manifold, $\mathcal{M}_y \in \mathcal{R}^{\mathrm{D}_y}$. This property provides an additional avenue for regularization: the predicted labels should be points drawn from (or near) a locally linear output manifold in the space of labels. Using the tangent space estimated by a small set of neighboring labels, we estimate the projection of a point on the output manifold by its tangent space representation, as shown in Figure 5.3.

Let $\mathcal{N}_y^i$ be the set of K_y nearest neighbors of label, $\hat{\mathbf{y}}_i \in \hat{\mathbf{Y}}$. The tangent space of $\hat{\mathbf{y}}_i$ is modeled using PCA to obtain the mean, $\mathbf{m}_i$, and the basis, $\mathbf{T}_i$, where $\mathbf{T}_i$ is a

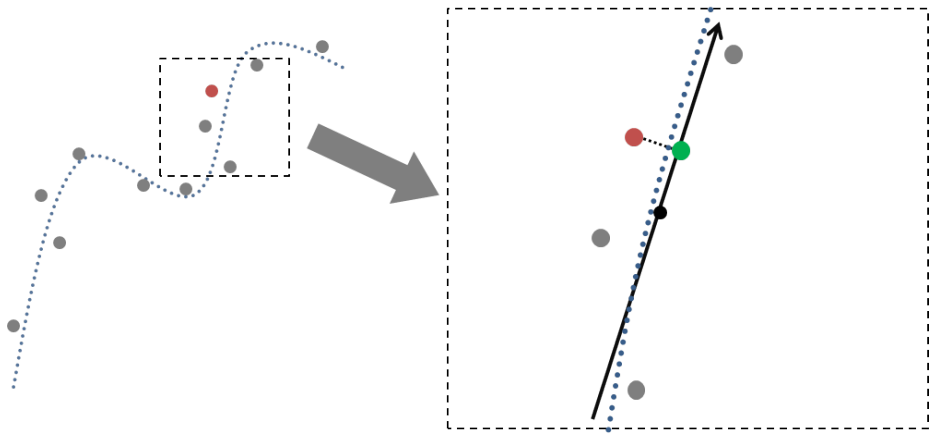


Figure 5.3: Illustration of label space regularization. For a given label (denoted by the red point), the regularizer encourages the predicted label to be close to the local tangent space (denoted by the green point).

matrix of size $d_y \times \mathrm{D}_y$. The projection of $\hat{\mathbf{y}}_i$ to the tangent space defined by $\mathbf{m}_i$, and $\mathbf{T}_i$ is $(\mathbf{T}_i)^\intercal \mathbf{T}_i(\hat{\mathbf{y}}_i - \mathbf{m}_i) + \mathbf{m}_i$. The output label manifold regularizer minimizes the difference between the predicted label and its reconstruction on the associated local tangent space:

$$\sum_{i=1}^{\mathrm{N}} ||\hat{\mathbf{y}}_i - (\mathbf{T}_i)^\intercal \mathbf{T}_i(\hat{\mathbf{y}}_i - \mathbf{m}_i) + \mathbf{m}_i||_2^2 \tag{5.3}$$

5.3.2 Optimization

Combining the image manifold regularization, label manifold regularization, and loss term, we have:

$$\begin{aligned} \underset{\hat{\mathbf{Y}}}{\operatorname{argmin}} \quad & \operatorname{Tr}(\hat{\mathbf{Y}}^\intercal \mathbf{H} \hat{\mathbf{Y}}) \\ & + \lambda_{\mathrm{s}} \sum_{i=1}^{\mathrm{N}} ||\hat{\mathbf{y}}_i - (\mathbf{T}_i)^\intercal \mathbf{T}_i(\hat{\mathbf{y}}_i - \mathbf{m}_i) - \mathbf{m}_i||_2^2 \\ & + \lambda_{\mathrm{l}} \operatorname{Tr}((\hat{\mathbf{Y}} - \mathbf{Y})^\intercal \breve{\mathbf{I}} (\hat{\mathbf{Y}} - \mathbf{Y})) \end{aligned} \tag{5.4}$$

where $\breve{\mathbf{I}}$ is a $\mathrm{N} \times \mathrm{N}$ diagonal matrix with a 1 at locations corresponding to originally labeled input, and 0 otherwise. This problem, with $\mathrm{N} \times \mathrm{D}_y$ unknown variables, is a

non-convex optimization due to the second term. Note that, $\mathbf{T}$ and $\mathbf{m}$, which depend on $\hat{\mathbf{Y}}$, cannot be expressed in a closed form. However, for fixed values of $\mathbf{m}$ and $\mathbf{T}$, the second term reduces to a quadratic function in terms of $\hat{\mathbf{Y}}$. Additionally, the Hessian operator, $\mathbf{H}$, is a positive semi-definite matrix, so this variant of Equation 5.4 becomes a quadratic function of $\hat{\mathbf{Y}}$ and can be solved efficiently. We solve Equation 5.4 using an alternating minimization approach, iterating between updating $\mathbf{m}$ and $\mathbf{T}$ and solving for $\hat{\mathbf{Y}}$. To initialize the method, we use only labeled examples for tangent space estimation. For each unlabeled example (e.g., $\mathbf{y}_i = \varnothing$), we assign $\mathbf{m}_i$ and $\mathbf{T}_i$ to be equal to the tangent space parameters of its nearest labeled neighbor in image space.

5.3.3 Algorithm

Given a set of images and associated labels for a subset of the input, our method, *Semi-Supervised Dual-Regularized Manifold Regression* (`SS-DRMR`), outlined in Algorithm 3, predicts multi-dimensional labels for the unlabeled examples.

Algorithm 3 `SS-DRMR`

Input: image features, $\mathbf{X}$; labels, $\mathbf{Y}$
Output: predicted labels, $\hat{\mathbf{Y}}$

Compute the global Hessian operator, $\mathbf{H}$
Estimate $\mathbf{m}$ and $\mathbf{T}$ using labeled examples
Solve for $\hat{\mathbf{Y}}_{(0)}$ (Equation 5.4)
$k \leftarrow 0$
repeat
    $k \leftarrow k + 1$
    With $\hat{\mathbf{Y}}_{(k-1)}$, estimate $\mathbf{m}$ and $\mathbf{T}$
    Solve for $\hat{\mathbf{Y}}_{(k)}$ (Equation 5.4)
until $||\hat{\mathbf{Y}}_{(k)} - \hat{\mathbf{Y}}_{(k-1)}||_F < \tau$ or $k = k_{\max}$
$\hat{\mathbf{Y}} \leftarrow \hat{\mathbf{Y}}_{(k)}$

For our method, the intrinsic dimensionality of the images and labels, d and d_y, respectively, and the neighborhood sizes, K and K_y, can be provided using prior

knowledge or estimated directly from the data. The algorithm terminates using one more user-specified criteria: the prediction changing by less than τ or $k_{\max}$ iterations. In Chapter 5.4, we describe the implementation details for `SS-DRMR`.

5.4 Experimental Evaluation

To evaluate `SS-DRMR` for semi-supervised multi-output regression, we compare the performance with the following methods for nonlinear regression:

- K-`NN` is a baseline approach where each unlabeled point is assigned the label corresponding to the average of its K nearest (labeled) neighbors in the input.
- Semi-supervised support vector regression (`SemiSVR`) [16], with the radial basis kernel, is a semi-supervised variant of SVR.
- Hessian semi-supervised regression (`HSSR`) [53] is a (scalar) regression method for nonlinear manifolds, also based on the Hessian regularizer.
- `KSPCA` [7] incorporates supervision with a nonlinear variant of PCA.

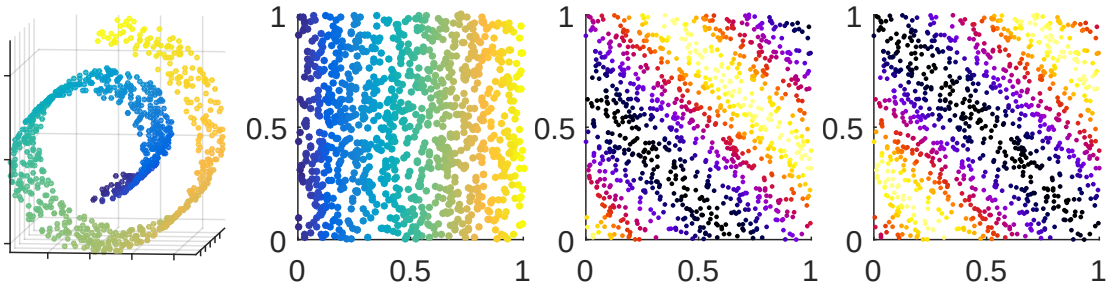
For all the experiments, the free parameters of each method (e.g., K for K-`NN`, kernel width for `SemiSVR` and `KSPCA`) are optimized using grid search and 5-fold cross-validation on the labeled input points. The fully supervised approaches were trained using only the labeled examples. For the scalar (single-output) methods, each dimension of the output was predicted independently.

We used labeled data sets with known ground truth. For each data set, the labels are normalized to $[0, 1]$.

- *Swiss Roll II* The Swiss Roll II data set (Figure 5.4) consists of 1,000 points randomly sampled from a 2D manifold embedded in 3D. For each 3D data point, $\mathbf{x}_i$, with 2D manifold coordinate, $\mathbf{z}_i = [z_{i1}\ z_{i2}]$, the label is defined as the cosine and sine of the sum of the input manifold coordinates, $\mathbf{y}_i = [\cos(z_{i1} + z_{i2})\ \sin(z_{i1} + z_{i2})]$.

- *Leaf Images* The Leaf Images data set (Figure 5.7) consists of 200 images simulating leaf growth. The images vary due to two types of variation: non-rigid leaf shape change and rotation; the set of images represent samples from an image manifold with an intrinsic dimensionality of 2. The labels for each image are three shape descriptors of the leaf: height (distance from stem to tip), width (distance from leftmost to rightmost tips), and area (number of foreground pixels).

For `SS-DRMR`, the intrinsic dimensionality of the image manifold, d, intrinsic dimensionality of the label manifold, d_y, and neighborhood sizes, K and K_y can be specified using prior knowledge of the data. However, we take a data-driven approach and directly estimate these parameters from the input. To estimate the intrinsic manifold dimensionality, we apply PCA on a neighborhood of 20 points from a small set of randomly selected examples and use the value corresponding to the "elbow point" of the residual variance curve. We found that the algorithm was robust to a large range of neighborhood sizes and set $K = 0.05\text{N}$ for both the input and output manifolds. For the regularization parameters, λ_s and λ_l, we use 5-fold cross validation to select values in the range $[10^{-4}, 10^{4}]$. For the termination criteria, we observed that for



(a) Input (3D) (b) Input (2D) (c) Label $\mathbf{y}^a$ (d) Label $\mathbf{y}^b$

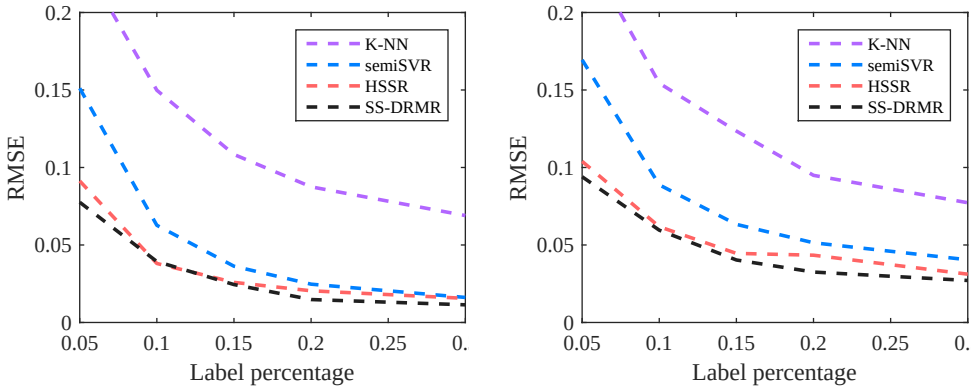
Figure 5.4: Swiss Roll II data with 2D labels. For the 2D label, $\mathbf{y} = [\mathbf{y}^a \ \mathbf{y}^b]$, the value of each dimension is indicated by the color of the points. For clarity, (b)-(d) are plotted using 2D manifold coordinates.

a range of values of ϵ, most experiments converged within 20 iterations, so we set $\tau = 10^{-2}$ and $k_{\max} = 20$. The algorithm is implemented in Matlab; on a standard PC, with an input of 1,000 samples, label set prediction takes less than 6 seconds on average.

5.4.1 Quantitative Evaluation

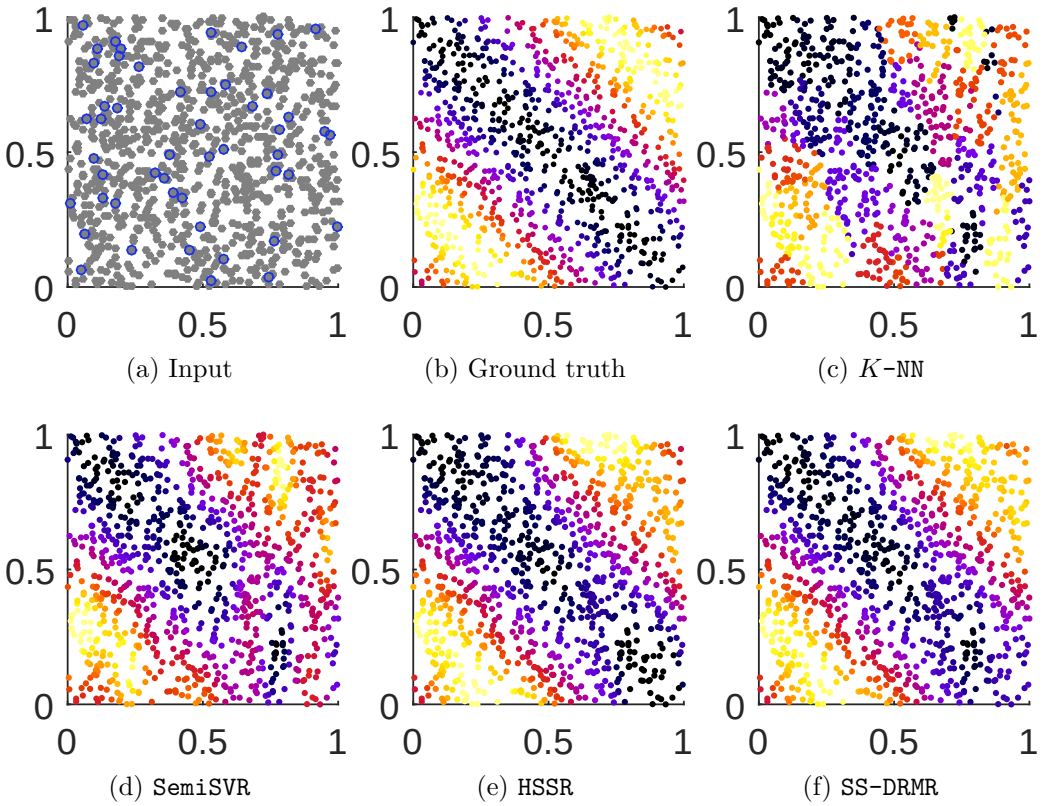
In this chapter, we evaluate the performance of each method by varying the percent of labeled examples and amount of label noise introduced. For these data sets, the values of each dimension of the labels are scaled to the range $[0, 1]$. Results are reported as the root mean squared error (RMSE) of the predicted labels on the unlabeled examples compared to ground truth.

Figure 5.6 shows results on the Swiss Roll II data set. Though the input is provided as the 3D ambient values, for ease of visualization, the graphs in Figure 5.6 are plotted using the 2D manifold coordinates. The value of each dimension of the label (Figure 5.6(c)) and (d)) is indicated by the color of the points. To test the performance of the methods, we varied the percentage of labeled input examples from 5% to 30%. In



(a) Clean (b) Noisy

Figure 5.5: For the Swiss Roll II data, each plot shows the RMSE of 10 repeated experiments.



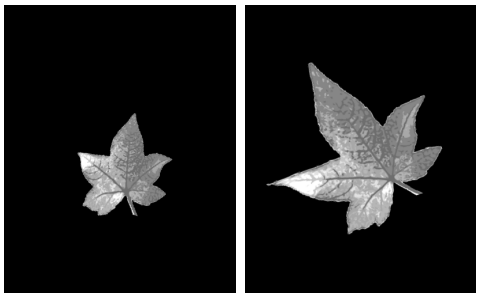
(a) Input (b) Ground truth (c) K-NN

(d) SemiSVR (e) HSSR (f) SS-DRMR

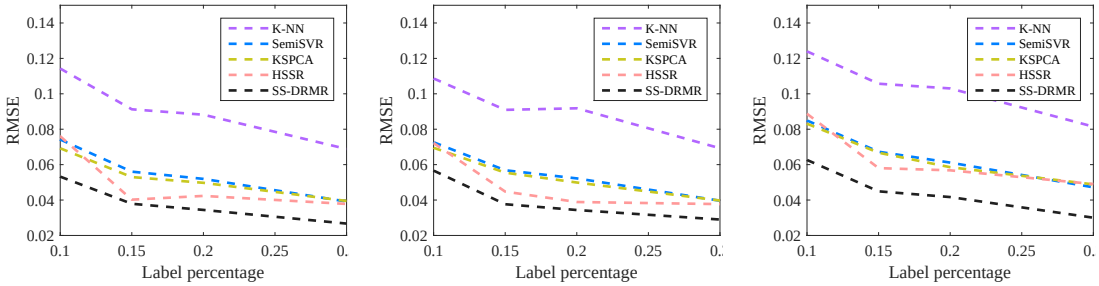
Figure 5.6: Results on Swiss Roll II data set with 5% labeled points. (a) shows the input with (randomly-selected) labeled examples denoted by blue circles. (b) shows the ground truth values (denoted by color) for one of the dimensions of the output labels. (c)-(f) show the output of each method for this dimension of the output. For clarity, results are plotted using 2D manifold coordinates.

a second variant, we added random Gaussian noise ($\sigma = 0.05$) to the provided labels. Figure 5.5 shows the results of these experiments reported as RMSE of the predicted label compared to the ground truth. For each setting, the results are averaged from 10 trials, where the subset of labeled points is selected randomly. Overall, with or without label noise, `SS-DRMR` has the lowest RMSE among all methods. Except for K-`NN`, the performance the methods converges as the number of labeled examples increases, and no noise is added. Additionally, the two methods based on Hessian regularization of this input space, `HSSR` and `SS-DRMR` performed similarly well at this prediction task on toy data. Figure 5.6 shows the estimated labels for an experiment with 20% of the input examples labeled. For K-`NN`, local patches of mislabeled examples can be observed. `SemiSVR` shows more global smoothness than K-`NN`, but overall pattern of the predicted labels is distorted when compared with the ground truth. Visually, the output of `HSSR` and `SS-DRMR` corresponds with low quantitative errors achieved across these experiments.

The Leaf images are used to evaluate the performance of multi-output regression on image manifolds. Similar to the Swiss Roll II experiment, we varied the percentage of labeled examples from 5% to 30% and added Gaussian noise ($\sigma = 0.05$) to each dimension of the labels. Figure 5.7 shows the prediction error for the Leaf Image data. All of the methods have decreasing prediction error as the percentage of labeled examples increases, with `SS-DRMR` being the top performer. This task is more challenging (e.g., high-dimensional image input, more complex labels) than the Swiss Roll II, and the results are consistent, except for the increased margin between `HSSR` and `SS-DRMR`. For more complex label spaces, dual regularization provides more



(a) Example images



(b) Width (c) Height (d) Area

Figure 5.7: Example input images and results from the Leaf Images experiment. Each plot shows the RMSE averaged over 10 trials for each dimension of the output label prediction task.

accurate predictions.

5.5 Applications

The experiments in Chapter 5.4 serve to compare `SS-DRMR` to related algorithms for multi-output prediction on synthetic data sets. However, many real-world image analysis problems, such as left ventricle segmentation and facial landmark detection, are instances of multi-output regression. In this chapter, we compare `SS-DRMR` to methods specifically designed for these tasks.

5.5.1 Facial Landmark Detection

Facial landmark detection facilitates, among other applications, head pose estimation and expression analysis. Most specialized approaches work best with high-

Table 5.1: RMSE (pixel units) of each method on the face data set.

Asthana2013	SemiSVR	HSSR	SS-DRMR
2.28	3.30	2.36	**1.96**

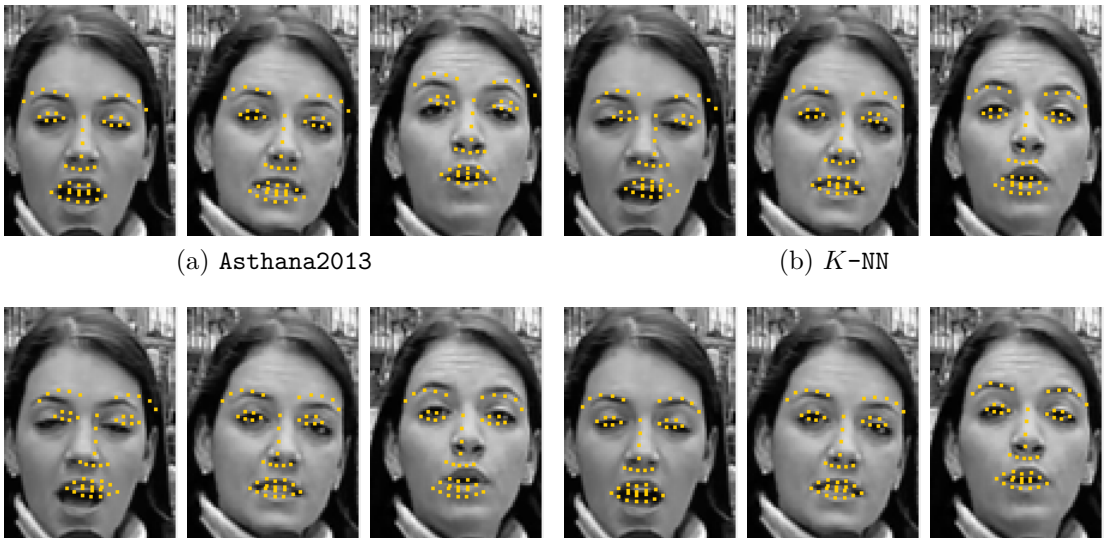
resolution images where facial features (eyes, nose, mouth) are distinct. For the task of facial landmark detection on small, low-resolution images, we compare methods for semi-supervised, multi-output regression to a specialized algorithm.

The input consists of 141 low-resolution images (100×72) from the YouTube Face Database [108]. Seven images (roughly 5% of the data) were randomly selected as the labeled examples. We compare SS-DRMR and other regression methods to a recent facial landmark detection algorithm (Asthana2013 [122, 5]). (This pre-trained method does not make use of the labeled input.) Performance was evaluated as the RMSE (in pixel units) of the predicted landmark location compared to the manually-annotated ground truth.

Table 5.1 gives the results for this task. For Asthana2013, for roughly 40% of the images, landmarks could not be detected, and no results were reported. The RMSE value only includes examples for which landmarks were reported. For the semi-supervised methods, including SS-DRMR, the results were computed from 10 repeated trials with different labeled input, and a prediction was provided for each input. SS-DRMR outperforms all other methods, including the task-specific algorithm. Figure 5.8 shows example frames with face detection results.

5.5.2 Left Ventricle Segmentation

One of the most common steps in pipelines for automated echocardiogram analysis is segmentation of the left ventricle. Compared to other imaging modalities, seg-



(a) `Asthana2013` (b) K-`NN`

(c) `SemiSVR` (d) `SS-DRMR`

Figure 5.8: Example results of facial landmark detection.

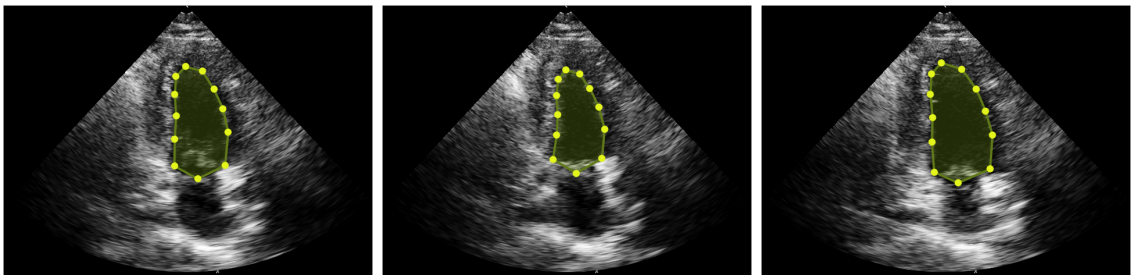


Figure 5.9: Three frames of the video used in the experiment. The ground truth labels consists of 11 points along the chamber wall.

mentation is complicated in ultrasound images due to speckle noise, which weakens the gradients near the boundaries of chamber walls and obscures the appearance of smaller cardiac structures. In terms of image manifold structured regression, both the frames of an echocardiogram video and the left ventricle contour lie on or near a cyclic manifold due to primary degree of freedom: cardiac motion.

The input is an apical four chamber (A4C) ultrasound video consisting of 180 frames (roughly 5 heart beats). The left ventricle contour is represented by the 2D image locations of 11 control points. For ground truth, each of the images was

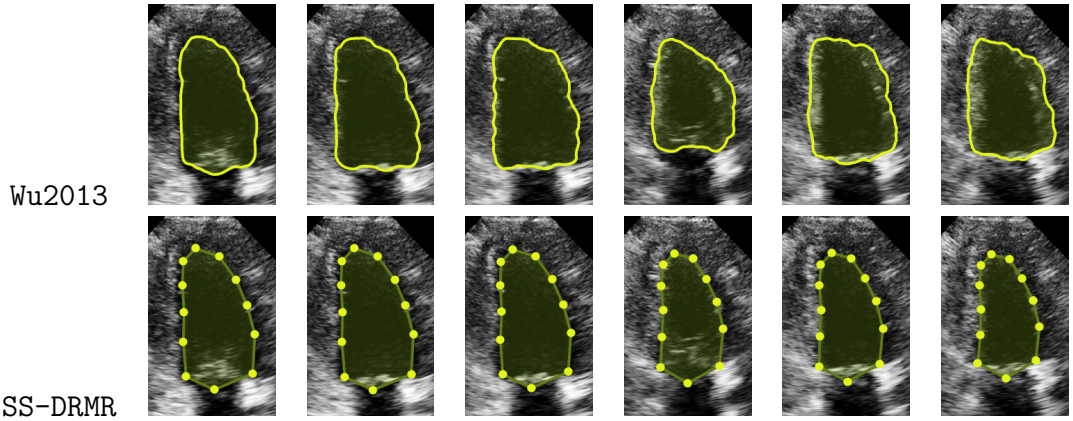


Figure 5.10: Representative segmentation results from SS-DRMR and a specialized active contour approach.

manually segmented, as shown in Figure 5.9. We compare our method to a recent image segmentation method (Wu2013 [113]), which is based on an adaptive diffusion flow active-contour model, and, for consecutive frames, initializes segmentation using the result from the previous frame. The free parameters were optimized to provide the best results. For both methods, five frames were provided as labeled input. (Four were selected randomly. Frame 1 was always included to bootstrap the Wu2013 method.)

Segmentation performance was evaluated using the Dice coefficient, which measures the overlap between the predicted segmentation and ground truth. For 10 trials (with different randomly-selected labeled example), the average Dice coefficient of SS-DRMR was 0.956, compared to 0.722 for Wu2013. Figure 5.10 shows representative results from both methods. This is a challenging segmentation task, with many images containing low-gradient edges around the left ventricle wall. The active contour approach tended to drift to other image regions with larger edge response.

5.6 Summary

We presented an algorithm for semi-supervised multi-output regression on image manifolds. We demonstrated that our dual-regularization approach outperforms competing methods on multi-output regression tasks, and without domain-specific tuning, our approach is competitive with recent, specialized algorithms for the tasks of facial landmark detection and left ventricle segmentation. The proposed approach only requires partial labels associated with a small subset of the images, which is most suitable in applications where multi-dimensional image labels require extensive manual effort or expertise to obtain. Although supervised multi-output problems (and in general, supervised structured output problems) have been intensively studied before, research on weakly supervised multi-output problems is still lacking. Our research presented in this chapter studies the semi-supervised setting of weakly supervised multi-output problems and is a variant of weakly supervised image manifold learning investigated in this dissertation.

CHAPTER 6: CONCLUSIONS

Given the availability of publicly-available images on the Web, as well as increasingly cheap sensors and storage, massive image sets are relatively easy to obtain. However, acquiring the associated image labels is a time-consuming manual job which tends to be less feasible with the drastic increase in the volume of image collections. Our research presented in this dissertation leverages weak supervision, which requires much less manual labeling effort than supervised methods, and incorporates the manifold structure of image sets to solve a range of computer vision problems. For weakly supervised manifold factorization, only image labels associated with the variation of interest are provided to remove unwanted image variation. For robust manifold regression, the semantically meaningful image labels are learned using noisy inputs obtained from automated algorithms. For semi-supervised multi-output manifold regression, multi-dimensional labels are estimated given only a small labeled set of images.

In summary, the research conducted in this dissertation is one of the first endeavors on weakly supervised learning on image manifolds, which aims to balance the surging amount of large-scale image sets and the difficulties in obtaining full, accurate annotations. By proposing a range of research problems with different application domains and different conditions of weak supervision, we hope to stimulate thoughts on future development of this underserved research area.

6.1 Future Work

There are several possible extensions of this work. First, our proposed algorithms can potentially be exploited as an efficient approach for image annotation. Given weak labels, the outputs of our algorithms are actually full image annotations that can be further used in subsequent fully supervised learning. For example, it would be interesting to investigate the ability of our methods to provide training data for deep neural networks.

Another direction is to investigate large-scale adaptations of the proposed algorithms. When applied to Internet-scale image collections, the image manifold structure can be extremely difficult to model by a small set of latent factors due to the innumerable amount of image sources and various kinds of image content. The problem is further complicated by the limit in computational power and memory. In this case, advanced computational models such as hierarchical decompositions of the manifold can be a possible solution.

Also, an interesting direction would be exploring the application of weakly supervised image manifold learning to more domains in the area of biomedical image and video analysis. Many images in clinical settings are collected along with quantitative measurements (e.g., ECG signals, respiratory volume, pulse rate, etc.). These metadata can potentially provide weak supervision to a variety of learning tasks with little or no manual labeling cost.

REFERENCES

[1] Vitaly Ablavsky and Stan Sclaroff. Learning parameterized histogram kernels on the simplex manifold for image and action classification. In *IEEE International Conference on Computer Vision*, pages 1473–1480, 2011.

[2] Sameer Agarwal, Yasutaka Furukawa, Noah Snavely, Ian Simon, Brian Curless, Steven M Seitz, and Richard Szeliski. Building rome in a day. In *IEEE International Conference on Computer Vision*, pages 72–79, 2009.

[3] Andreas Alfons, Christophe Croux, Sarah Gelper, et al. Sparse least trimmed squares regression for analyzing high-dimensional large data sets. *The Annals of Applied Statistics*, 7(1):226–248, 2013.

[4] Yasemin Altun, Mikhail Belkin, and David A Mcallester. Maximum margin semi-supervised learning for structured variables. In *Advances in neural information processing systems*, pages 33–40, 2005.

[5] Akshay Asthana, Stefanos Zafeiriou, Shiyang Cheng, and Maja Pantic. Robust discriminative response map fitting with constrained local models. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 3444–3451, 2013.

[6] David Atkinson, Michael Burcher, Jerome Declerck, and J Alison Noble. Respiratory motion compensation for 3-d freehand echocardiography. *Ultrasound in Medicine and Biology*, 27(12):1615 – 1620, 2001.

[7] Elnaz Barshan, Ali Ghodsi, Zohreh Azimifar, and Mansoor Zolghadri Jahromi. Supervised principal component analysis: visualization, classification and regression on subspaces and submanifolds. *Pattern Recognition*, 44(7):1357–1371, 2011.

[8] Mikhail Belkin, Partha Niyogi, and Vikas Sindhwani. Manifold regularization: A geometric framework for learning from labeled and unlabeled examples. *The Journal of Machine Learning Research*, 7:2399–2434, 2006.

[9] Chiraz BenAbdelkader. Robust head pose estimation using supervised manifold learning. In *European Conference on Computer Vision*, pages 518–531, 2010.

[10] Christopher M Bishop et al. *Pattern recognition and machine learning*, volume 4. springer New York, 2006.

[11] Matthew Brand. Charting a manifold. In *Advances in neural information processing systems*, pages 961–968, 2002.

[12] Ignas Budvytis, Vijay Badrinarayanan, and Roberto Cipolla. Semi-supervised video segmentation using tree structured graphical models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(11):2751–2764, Nov 2013.

[13] Chris Buehler, Michael Bosse, and Leonard McMillan. Non-metric image-based rendering for video stabilization. In *IEEE Conference on Computer Vision and Pattern Recognition*, volume 2, pages 609–614, 2001.

[14] Deng Ca, Xiaofei He, Kun Zhou, Jiawei Han, and Hujun Bao. Locality sensitive discriminant analysis. In *International Joint Conferences on Artificial Intelligence*, pages 708–713, 2007.

[15] Deng Cai, Xiaofei He, and Jiawei Han. Semi-supervised discriminant analysis. In *IEEE International Conference on Computer Vision*, pages 1–7, 2007.

[16] Gustavo Camps-Valls, Jordi Muñoz-Marí, Luis Gómez-Chova, Katja Richter, and Javier Calpe-Maravilla. Biophysical parameter estimation with a semisupervised support vector machine. *IEEE Geoscience and Remote Sensing Letters*, 6(2):248–252, April 2009.

[17] Chih-Chung Chang and Chih-Jen Lin. LIBSVM: a library for support vector machines. *ACM Transactions on Intelligent Systems and Technology*, 2(3):27, 2011.

[18] Olivier Chapelle, Bernhard Schlkopf, and Alexander Zien. Semi-supervised learning. *MIT Press*, 2006.

[19] Muhammad Shahzad Cheema, Abdalrahman Eweiwi, and Christian Bauckhage. Human activity recognition by separating style and content. *Pattern Recognition Letters*, pages 130–138, 2013.

[20] Reuven Cohen and Liran Katzir. The generalized maximum coverage problem. *Information Processing Letters*, 108(1):15–22, 2008.

[21] Timothy F Cootes, Carole J Twining, Vladimir S Petrović, Kolawole O Babalola, and Christopher J Taylor. Computing accurate correspondences across groups of images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 32(11):1994–2005, 2010.

[22] Mark Cox, Sridha Sridharan, Simon Lucey, and Jeffrey Cohn. Least squares congealing for unsupervised alignment of images. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–8, June 2008.

[23] Mark Cox, Sridha Sridharan, Simon Lucey, and Jeffrey Cohn. Least-squares congealing for large numbers of images. In *IEEE International Conference on Computer Vision*, pages 1949–1956, Sept 2009.

[24] Navneet Dalal and Bill Triggs. Histograms of oriented gradients for human detection. In *IEEE Conference on Computer Vision and Pattern Recognition*, volume 1, pages 886–893, 2005.

[25] Dick de Ridder, Olga Kouropteva, Oleg Okun, Matti Pietikäinen, and Robert PW Duin. Supervised locally linear embedding. In *Artificial Neural Networks and Neural Information Processing*, pages 333–341, 2003.

[26] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 248–255, 2009.

[27] David L Donoho and Carrie Grimes. Hessian eigenmaps: Locally linear embedding techniques for high-dimensional data. *Proceedings of the National Academy of Sciences*, 100(10):5591–5596, 2003.

[28] Nicolas Duchateau, Mathieu De Craene, Gemma Piella, Etelvino Silva, Adelina Doltra, Marta Sitges, Bart H Bijnens, and Alejandro F Frangi. A spatiotemporal statistical atlas of motion for the quantification of abnormal myocardial tissue velocities. *Medical image analysis*, 15(3):316–328, 2011.

[29] Olivier Duchenne, Ivan Laptev, Josef Sivic, Francis Bach, and Jean Ponce. Automatic annotation of human actions in video. In *IEEE International Conference on Computer Vision*, pages 1491–1498, 2009.

[30] Yonina C Eldar and Gitta Kutyniok. *Compressed sensing: theory and applications*. Cambridge University Press, 2012.

[31] Ahmed Elgammal, Ramani Duraiswami, David Harwood, and Larry S Davis. Background and foreground modeling using nonparametric kernel density estimation for visual surveillance. *Proceedings of the IEEE*, 90(7):1151–1163, 2002.

[32] Ahmed Elgammal and Chan-Su Lee. Separating style and content on a nonlinear manifold. In *IEEE Computer Conference on Computer Vision and Pattern Recognition*, pages 478–485, 2004.

[33] Rob Fergus, Yair Weiss, and Antonio Torralba. Semi-supervised learning in gigantic image collections. In *Advances in Neural Information Processing Systems*, pages 522–530, 2009.

[34] Martin A Fischler and Robert C Bolles. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Communications of the ACM*, 24(6):381–395, 1981.

[35] Jacob Gardner, Matt Kusner, Kilian Weinberger, John Cunningham, et al. Bayesian optimization with inequality constraints. In *International Conference on Machine Learning*, pages 937–945, 2014.

[36] Dian Gong and Gerard Medioni. Dynamic manifold warping for view invariant action recognition. In *IEEE International Conference on Computer Vision*, pages 571–578, 2011.

[37] Dian Gong, Fei Sha, and Gérard G Medioni. Locally linear denoising on image manifolds. In *International Conference on Artificial Intelligence and Statistics*, pages 265–272, 2010.

[38] Joger Hansegard, Stig Urheim, Ketil Lunde, and Stein Inge Rabben. Constrained active appearance models for segmentation of triplane echocardiograms. *IEEE Transactions on Medical Imaging*, 26(10):1391–1400, Oct 2007.

[39] Per Christian Hansen. Analysis of discrete ill-posed problems by means of the l-curve. *Society for Industrial and Applied Mathematics Review*, 34(4):561–580, 1992.

[40] Glenn Hartmann, Matthias Grundmann, Judy Hoffman, David Tsai, Vivek Kwatra, Omid Madani, Sudheendra Vijayanarasimhan, Irfan Essa, James Rehg, and Rahul Sukthankar. Weakly supervised learning of object segmentations from web-scale video. In *European Conference on Computer Vision Workshops and Demonstrations*, pages 198–208. Springer, 2012.

[41] Dennis M Healy and Gustavo K Rohde. Fast global image registration using random projections. In *International Symposium on Biomedical Imaging*, pages 476–479. IEEE, 2007.

[42] Tobias Heimann and Hans-Peter Meinzer. Statistical shape models for 3d medical image segmentation: A review. *Medical Image Analysis*, 13(4):543 – 563, 2009.

[43] Matthias Hein and Markus Maier. Manifold denoising. In *Advances in neural information processing systems*, pages 561–568, 2006.

[44] Matthias Hein and Markus Maier. Manifold denoising as preprocessing for finding natural representations of data. In *AAAI Conference on Artificial Intelligence*, pages 1646–1649, 2007.

[45] Minh Hoai and Fernando De la Torre. Max-margin early event detectors. *International Journal of Computer Vision*, 107(2):191–202, 2014.

[46] Jenq-Neng Hwang, S-R Lay, and Alan Lippman. Nonparametric multivariate density estimation: a comparative study. *IEEE Transactions on Signal Processing*, 42(10):2795–2810, 1994.

[47] Mohammad Islam, Nathan Jacobs, Hui Wu, and Richard Souvenir. Images+weather: Collection, validation, and refinement. In *IEEE Conference on Computer Vision and Pattern Recognition Workshop on Ground Truth*, 2013.

[48] Mohammad Islam, Scott Workman, Hui Wu, Richard Souvenir, and Nathan Jacobs. Exploring the geo-dependence of human face appearance. In *IEEE Winter Conference on Applications of Computer Vision*, 2014.

[49] Nathan Jacobs, Nathaniel Roman, and Robert Pless. Consistent temporal variations in many outdoor scenes. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–6, 2007.

[50] Jørgen Arendt Jensen. Field: a program for simulating ultrasound systems. In *10th Nordic-Baltic Conference on Biomedical Imaging*, volume 34, pages 351–353, 1996.

[51] Jørgen Arendt Jensen and Niels Bruun Svendsen. Calculation of pressure fields from arbitrarily shaped, apodized, and excited ultrasound transducers. *IEEE Transactions on Ultrasonics, Ferroelectrics and Frequency Control*, 39(2):262–267, 1992.

[52] Ian Jolliffe. *Principal Component Analysis*. Springer Series in Statistics. Springer, 2002.

[53] Kwang I Kim, Florian Steinke, and Matthias Hein. Semi-supervised regression using hessian energy with an application to semi-supervised dimensionality reduction. In *Advances in Neural Information Processing Systems*, pages 979–987, 2009.

[54] Seung-Jean Kim, Kwangmoo Koh, Michael Lustig, Stephen Boyd, and Dimitry Gorinevsky. An interior-point method for large-scale l1-regularized least squares. *IEEE Journal of Selected Topics in Signal Processing*, 1(4):606–617, 2007.

[55] Neeraj Kumar, Alexander C Berg, Peter N Belhumeur, and Shree K Nayar. Attribute and simile classifiers for face verification. In *IEEE International Conference on Computer Vision*, pages 365–372, Oct 2009.

[56] Erik G Learned-Miller. Data driven image models through continuous joint alignment. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 28(2):236–250, Feb 2006.

[57] Ken-Yi Lee, Yung-Yu Chuang, Bing-Yu Chen, and Ming Ouhyoung. Video stabilization using robust feature trajectories. In *IEEE International Conference on Computer Vision*, pages 1397–1404, 2009.

[58] KY Leung, Mikhail G Danilouchkine, Marijn van Stralen, Nico de Jong, Antonius FW van der Steen, and Johan G Bosch. Probabilistic framework for tracking in artifact-prone 3d echocardiograms. *Medical image analysis*, 14(6):750–758, 2010.

[59] Bo Li, Chun-Hou Zheng, and De-Shuang Huang. Locally linear discriminant embedding: An efficient method for face recognition. *Pattern Recognition*, 41(12):3813 – 3821, 2008.

[60] Wei Liu, Jun Wang, and Shih-Fu Chang. Robust and scalable graph-based semisupervised learning. *Proceedings of the IEEE*, 100(9):2624–2638, 2012.

[61] Dijun Luo and Heng Huang. Video motion segmentation using new adaptive manifold denoising model. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 65–72, June 2014.

[62] Michael Lustig, David Donoho, and John M Pauly. Sparse mri: The application of compressed sensing for rapid mr imaging. *Magnetic resonance in medicine*, 58(6):1182–1195, 2007.

[63] Yasuyuki Matsushita, Eyal Ofek, Weina Ge, Xiaoou Tang, and Heung-Yeung Shum. Full-frame video stabilization with motion inpainting. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 28(7):1150–1163, July 2006.

[64] Ha Q Minh and Vikas Sindhwani. Vector-valued manifold regularization. In *Proceedings of International Conference on Machine Learning*, pages 57–64, 2011.

[65] Anurag Mittal and Nikos Paragios. Motion-based background subtraction using adaptive kernel density estimation. In *IEEE Conference on Computer Vision and Pattern Recognition*, volume 2, pages 302–309, 2004.

[66] Carlos Morimoto and Rama Chellappa. Evaluation of image stabilization algorithms. In *IEEE International Conference on Acoustics, Speech and Signal Processing*, volume 5, pages 2789–2792, 1998.

[67] Jacinto C Nascimento and Gustavo Carneiro. Top-down segmentation of non-rigid visual objects using derivative-based search on sparse manifolds. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 1963–1970, 2013.

[68] Jacinto C. Nascimento and Gustavo Carneiro. Non-rigid segmentation using sparse low dimensional manifolds and deep belief networks. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 288–295, June 2014.

[69] Nagarajan Natarajan, Inderjit Dhillon, Pradeep Ravikumar, and Ambuj Tewari. Learning with noisy labels. In *Advances in Neural Information Processing Systems*, pages 1196–1204, 2013.

[70] George L Nemhauser, Laurence A Wolsey, and Marshall L Fisher. An analysis of approximations for maximizing submodular set functions. *Mathematical Programming*, 14(1):265–294, 1978.

[71] J Alison Noble and Djamal Boukerroui. Ultrasound image segmentation: a survey. *IEEE Transactions on Medical Imaging*, 25(8):987–1010, Aug 2006.

[72] Jooyoung Park and Irwin W Sandberg. Universal approximation using radial-basis-function networks. *Neural computation*, 3(2):246–257, 1991.

[73] Alonso Patron-Perez, Marcin Marszalek, Ian Reid, and Andrew Zisserman. Structured learning of human interactions in tv shows. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 34(12):2441–2453, Dec 2012.

[74] Yigang Peng, A Ganesh, J. Wright, Wenli Xu, and Yi Ma. Rasl: Robust alignment by sparse and low-rank decomposition for linearly correlated images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 34(11):2233–2246, Nov 2012.

[75] Robert Pless and Ian Simon. Using thousands of images of an object. In *International Conference on Computer Vision, Pattern Recognition and Image Processing*, 2002.

[76] Chen Qian, Xiao Sun, Yichen Wei, Xiaoou Tang, and Jian Sun. Realtime and robust hand tracking from depth. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 1106–1113, 2014.

[77] Rahul Raguram, Ondrej Chum, Marc Pollefeys, Jose Matas, and Jens Frahm. Usac: A universal framework for random sample consensus. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(8):2022–2038, Aug 2013.

[78] Vikas C Raykar and Shipeng Yu. Ranking annotators for crowdsourced labeling tasks. In *Advances in neural information processing systems*, pages 1809–1817, 2011.

[79] Peter J Rousseeuw. Least median of squares regression. *Journal of the American statistical association*, 79(388):871–880, 1984.

[80] Sam T Roweis and Lawrence K Saul. Nonlinear dimensionality reduction by locally linear embedding. *Science*, 290(5500):2323–2326, 2000.

[81] Carlos Santiago, Jacinto C Nascimento, and Jorge S Marques. 3d left ventricular segmentation in echocardiography using a probabilistic data association deformable model. In *IEEE Conference on Image Processing*, pages 606–610, 2013.

[82] Chunhua Shen, Guosheng Lin, and A. van den Hengel. Structboost: Boosting methods for predicting structured output variables. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 36(10):2089–2103, Oct 2014.

[83] Noah Snavely, Steven M Seitz, and Richard Szeliski. Modeling the world from internet photo collections. *International Journal of Computer Vision*, 80(2):189–210, 2008.

[84] Jasper Snoek, Hugo Larochelle, and Ryan P Adams. Practical bayesian optimization of machine learning algorithms. In *Advances in Neural Information Processing Systems*, pages 2951–2959, 2012.

[85] Aristeidis Sotiras, Christos Davatzikos, and Nikos Paragios. Deformable medical image registration: A survey. *IEEE Transactions on Medical Imaging*, 32(7):1153–1190, 2013.

[86] Žiga Špiclin, Boštjan Likar, and Franjo Pernuš. Groupwise registration of multimodal images by an efficient joint entropy minimization scheme. *IEEE Transactions on Image Processing*, 21(5):2546–2558, May 2012.

[87] Michael Suhling, Muthuvel Arigovindan, Christian Jansen, Patrick Hunziker, and Michael Unser. Myocardial motion analysis from b-mode echocardiograms. *IEEE Transactions on Image Processing*, 14(4):525–536, 2005.

[88] Jinhui Tang, Shuicheng Yan, Richang Hong, Guo-Jun Qi, and Tat-Seng Chua. Inferring semantic concepts from community-contributed images and noisy tags. In *Proceedings of ACM international conference on Multimedia*, pages 223–232, 2009.

[89] Peter C Tay, Christopher D Garson, Scott T Acton, John Hossack, et al. Ultrasound despeckling for contrast enhancement. *IEEE Transactions on Image Processing*, 19(7):1847–1860, 2010.

[90] Joshua B Tenenbaum, Vin De Silva, and John C Langford. A global geometric framework for nonlinear dimensionality reduction. *Science*, 290(5500):2319–2323, 2000.

[91] Joshua B Tenenbaum and William T Freeman. Separating style and content with bilinear models. *Neural computation*, 12(6):1247–1283, 2000.

[92] Ioannis Tsochantaridis, Thorsten Joachims, Thomas Hofmann, and Yasemin Altun. Large margin methods for structured and interdependent output variables. In *Journal of Machine Learning Research*, pages 1453–1484, 2005.

[93] Laurens JP van der Maaten, Eric O Postma, and H Jaap van den Herik. Dimensionality reduction: a comparative review. *Tilburg University Technical Report*, 2009.

[94] Bo Wang and Zhuowen Tu. Sparse subspace denoising for image manifolds. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 468–475, 2013.

[95] Chang Wang and Sridhar Mahadevan. Manifold alignment using procrustes analysis. In *Proceedings of international conference on Machine learning*, pages 1120–1127. ACM, 2008.

[96] Chang Wang and Sridhar Mahadevan. Manifold alignment without correspondence. In *Proceedings of international jont conference on Artifical intelligence*, pages 1273–1278. Morgan Kaufmann Publishers Inc., 2009.

[97] Chang Wang and Sridhar Mahadevan. Manifold alignment preserving global geometry. In *Proceedings of international joint conference on Artificial Intelligence*, pages 1743–1749, 2013.

[98] Jack M Wang, David J Fleet, and Aaron Hertzmann. Multifactor gaussian process models for style-content separation. In *international conference on Machine learning*, pages 975–982. ACM, 2007.

[99] Jun Wang, Sanjiv Kumar, and Shih-Fu Chang. Semi-supervised hashing for scalable image retrieval. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 3424–3431, 2010.

[100] Li Wang, Michael D Gordon, and Ji Zhu. Regularized least absolute deviations regression and an efficient algorithm for parameter tuning. In *IEEE International Conference on Data Mining*, pages 690–700, 2006.

[101] Ruiping Wang, Shiguang Shan, Xilin Chen, Jie Chen, and Wen Gao. Maximal linear embedding for dimensionality reduction. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 33(9):1776–1792, 2011.

[102] Weiran Wang, Zhengdong Lu, and Miguel Á. Carreira-Perpiñán. A denoising view of matrix completion. In *Advances in Neural Information Processing Systems*, pages 334–342, 2011.

[103] Xianwang Wang and Ruigang Yang. Learning 3d shape from a single facial image via non-linear manifold embedding and alignment. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 414–421, June 2010.

[104] Zhou Wang, Alan C Bovik, Hamid R Sheikh, and Eero P Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE Transactions on Image Processing*, 13(4):600–612, 2004.

[105] Kilian Q Weinberger, Fei Sha, and Lawrence K Saul. Learning a kernel matrix for nonlinear dimensionality reduction. In *Proceedings of International Conference on Machine Learning*, pages 839–846, Banff, Canada, 2004.

[106] Daniel Weinland, Remi Ronfard, and Edmond Boyer. Free viewpoint action recognition using motion history volumes. *Computer Vision and Image Understanding*, 104(2):249–257, 2006.

[107] Christian Wengert, Matthijs Douze, and Hervé Jégou. Bag-of-colors for improved image search. In *Proceedings of ACM international conference on Multimedia*, pages 1437–1440, 2011.

[108] Lior Wolf, Tal Hassner, and Itay Maoz. Face recognition in unconstrained videos with matched background similarity. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 529–534, 2011.

[109] John Wright, Allen Y Yang, Arvind Ganesh, Shankar S Sastry, and Yi Ma. Robust face recognition via sparse representation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 31(2):210–227, 2009.

[110] Hui Wu, Dustin M. Bowers., Toan T. Huynh, and Richard Souvenir. Biomedical video denoising using supervised manifold learning. In *IEEE International Symposium on Biomedical Imaging*, pages 1244–1247, 2013.

[111] Hui Wu, Dustin M Bowers, Toan T Huynh, and Richard Souvenir. Deformable alignment using random projections of landmark images. In *IEEE International Symposium on Biomedical Imaging*, pages 754–757, April 2014.

[112] Hui Wu, Toan T. Huynh, and Richard Souvenir. Echocardiogram enhancement using supervised manifold denoising. *Medical Image Analysis*, 2015.

[113] Yuwei Wu, Yuanquan Wang, and Yunde Jia. Adaptive diffusion flow active contours for image segmentation. *Computer Vision and Image Understanding*, 117(10):1421–1435, 2013.

[114] Shuicheng Yan, Huan Wang, Yun Fu, Jun Yan, Xiaoou Tang, and T.S. Huang. Synchronized submanifold embedding for person-independent pose estimation and beyond. *IEEE Transactions on Image Processing*, 18(1):202–210, Jan 2009.

[115] Shihui Ying, Guorong Wu, Qian Wang, and Dinggang Shen. Groupwise registration via graph shrinkage on the image manifold. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 2323–2330, 2013.

[116] Guoxian Yu, Guoji Zhang, Carlotta Domeniconi, Zhiwen Yu, and Jane You. Semi-supervised classification based on random subspace dimensionality reduction. *Pattern Recognition*, 45(3):1119–1135, 2012.

[117] Yong Yue, John W Clark, and Dirar S Khoury. Speckle tracking in intracardiac echocardiography for the assessment of myocardial deformation. *IEEE Transactions on Biomedical Engineering*, 56(2):416–425, 2009.

[118] Yong Yue, Mihai M Croitoru, Akhil Bidani, Joseph B Zwischenberger, and John W Clark. Nonlinear multiscale wavelet diffusion for speckle suppression and edge enhancement in ultrasound images. *IEEE Transactions on Medical Imaging*, 25(3):297–311, 2006.

[119] Qinghua Zhang, Alfredo Illanes Manriquez, Claire Médigue, Yves Papelier, and Michel Sorine. An algorithm for robust and efficient location of t-wave ends in electrocardiograms. *IEEE Transactions on Biomedical Engineering*, 53(12):2544 –2552, Dec 2006.

[120] Shaohua Kevin Zhou. Shape regression machine and efficient segmentation of left ventricle endocardium from 2d b-mode echocardiogram. *Medical Image Analysis*, 14(4):563 – 581, 2010.

[121] Jun Zhu, Junhua Mao, and Alan L Yuille. Learning from weakly supervised data by the expectation loss svm (e-svm) algorithm. In *Advances in Neural Information Processing Systems*, pages 1125–1133, 2014.

[122] Xiangxin Zhu and Deva Ramanan. Face detection, pose estimation, and landmark localization in the wild. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 2879–2886, 2012.

[123] Xiaojin Zhu. Semi-supervised learning literature survey. *Computer Sciences Technical Report 1530, University of Wisconsin-Madison*, 2005.