Object Segmentation and Tracking in 3D Video With Sparse Depth Information Using a Fully Connected CRF Model

Ido Ofir
Computer Science Department
Stanford University

December 17, 2011

Abstract

This project extends the most state-of-the-art technique for multi-class image segmentation [1] to a technique that incorporates geometric and temporal data for multi-class segmentation and tracking of unknown classes of objects. Previous video rate segmentation techniques only achieve fast performance with considerably lower accuracy by reducing the density of pairwise connected graphs. This has been done using hierarchical region-level models or pixel-level models with connectivity limited to nearest neighbors. A fully connected CRF model over the complete set of all pixels results in billions of connections and has therefore been impractical to segment before now. Using this new technique which defines edge potentials as a linear combination of Gaussian Kernels, it is possible to approximate the segmentation of a fully connected CRF in a fraction of a second. This project extends the core algorithm to incorporate sparse geometric features from a laser scanner into the segmentation. It also provides some tracking capabilities for the segmented objects.

1 Introduction

Basic CRF models are composed of unary potentials on individual pixels or image patches and pairwise potentials on neighboring pixels or patches. New state-of-the-art techniques allow for a highly efficient inference algorithm for fully connected CRF models in which the pairwise edge potentials are dened by a linear combination of Gaussian kernels in an arbitrary feature space.

In the problem this paper considers that feature space included color and geometry information from a camera and depth sensor. The data sequence comprised of a long series of frames from video recordings, with the purpose of segmenting and tracking primary objects in each video. The problem was augmented by the sparse nature of the geometric information. Previous CRF models were designed to resolve a data set with consistent features. Due to the physical limitations of range sensors, geometric features could not be measured for 10%-30% of indoor data and 30%-90% of outdoor data. This inconsistency in the features available required considerable changes to the CRF model.

1.1 The Fully Connected CRF Model

The fully connected CRF model that this project extends solves an energy function where the potential for any given edge in the graph is given by a linear combination of unary and binomial compatibility scores. The unary potential in this approach computes the probability that any individual pixel has a specific labeling. This learned probability is easily learned and computed in O(n) time, but is insufficiently accurate for labeling. The pairwise potential is based on the expectation that two pixels share the same label. Two pixels with different labels accrue a penalty that is determined by the compatibility of the labels. This allows for labels the often occur together to receive smaller penalties. The brilliance of this technique lies in the fast mean field approximation of the distribution the minimizes the KL-divergence in linear time with respect to the number of features.

1.2 Mean Field Approximation

Instead of computing the exact distribution P(X), the mean eld approximation computes a distribution Q(X) that minimizes the KL-divergence. Each iteration of the algorithm performs three steps.

1. A message passing step where each variable evaluates a sum over all other variables.

2. A compatibility transform. O(n)

3. A local update passing. O(n)

The message passing step can be approximated using the Permutohedral Lattice, which reduces the quadratic complexity to linear complexity.

1.3 The Permutohedral Lattice

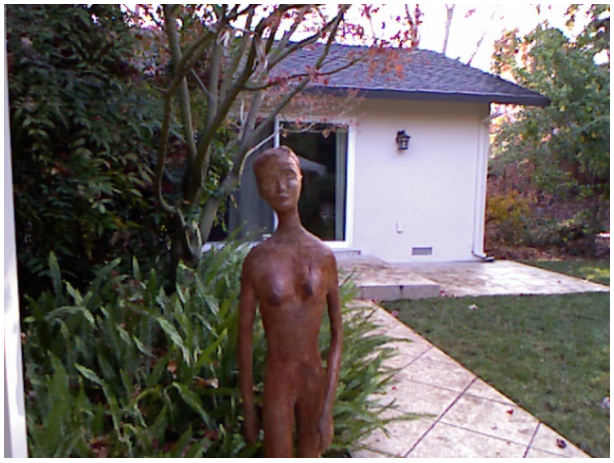
The Permutohedral Lattice [2] provides a high-dimensional Gaussian lter that is both linear in input size and polynomial in dimensionality. The d-dimensional permutohedral lattice is formed by projecting the scaled grid $(d+1)Z^{d+1}$ onto the plane $\vec{x} \times \vec{1} = 0$. More on the Permutohedral Lattice can be found here:

`http:\\graphics.stanford.edu/papers/permutohedral`

2 Data

This project used data from two distinct sources. It was the original intent of the project to focus exclusively on data from the Stanford Autonomous Car. After much analysis it proved that the data acquired by that system was too sparse (10% available) due to the laser scanner. The rest of the project, including the final result was done with the Xbox kinect. The images below show a typical frame captured from the kinect. The image on the right shows the depth data that was successfully measured. note the measurement error in the mid-right portion of the frame which greatly affected the performance of the segmentation.

2.1 Depth Sensors



2.2 Spares Data

Sparse data was problematic for several reasons but the greatest of which was the gaussian kernels for pairwise potentials. Where data was missing in one of the two samples the potential energy would discount them as too dissimilar. Where both data samples showed missing data the kernel would assign them a very strong affinity and would not label them as separate classes.

3 Implementation

The solution used in this project was to substitute the kernel for missing data points with an alternative kernel that would provide the correct affinity score using the remaining features. The whole project was implemented in C++ with the exception of the Mixture of Gaussians unary features that where borrowed from MATLAB.

3.1 Unary Classifiers

Of the several simple unary features used in this project the best performing one I tried was the mixture of gaussians algorithm. The data I used was completely unsupervised which limited my options in this respect. It is important to note that the performance of the unary classifier was directly and linearly related to the performance of the segmentation overall. This observation can be explained by considering the roles of unary and bilateral potentials in the CRF model. The Unary features attempt to label pixels as different classes while the bilateral potentials attempt to minimize the classes labeled. Without a proper labeling of the unary feature the algorithm reduced the segmentation to a single class.

3.1.1 Mixture of Gaussians

I used mixture of gaussians to label the pixels into 7 different classes and used the labelings as initial probabilities for each class. The mixture of gaussians segmentation performed much better on color features without the geometry information. The number of labels and potential weights used throughout the project was selected to best match the data using grid search.

3.1.2 Previous Segmentation Cost of Change

The final unary features were a linear combination of the mixture of gaussians component and the probabilities derived from the labelings of the previous frame. This was done for several reasons, chief of which was to allow for tracking in completely unsupervised segmentation. The assumption behind this approach is that in a high frame rate video, there is a high likelihood that a pixel will maintain its classification from the previous frame. This approach work well in practice, but is susceptible to drift errors. The weights for the probabilities and for the coefficients of the linear combination where tuned by hand.

3.2 Pairwise Cost Functions

The original intent of the project was to relay of pure, real world geometry and color features to form the gaussian kernels for the pairwise potential of the CRF. However, due to the issues already discussed, it proved impossible to relay on these kernels alone in the absence of valid geometric data.

3.2.1 Alternate Kernels in Image Space

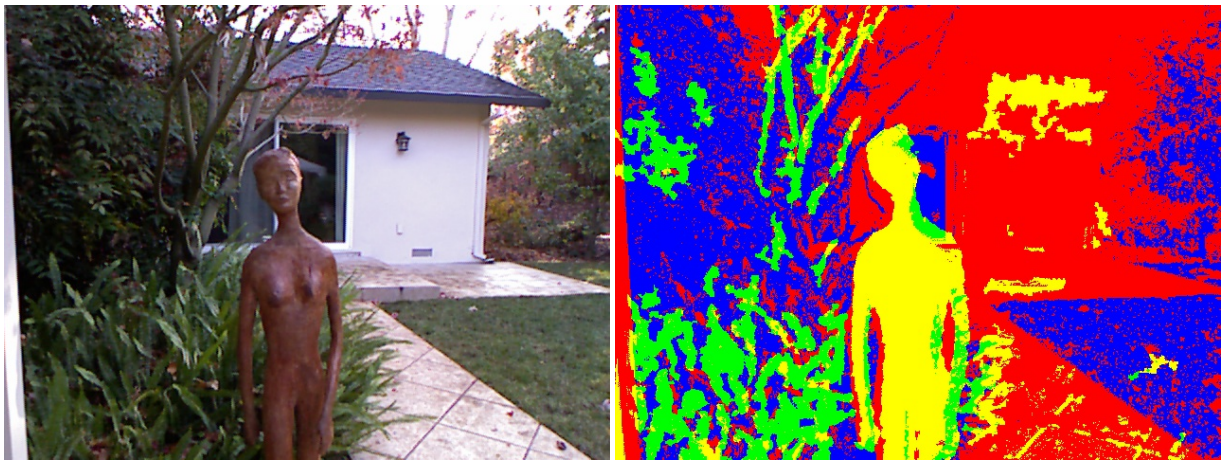
Even in the absence of any depth measurement, some geometric data is still available. The Alternative kernels I used as a replacement for samples with partial data, were the original image space appearance kernels. These gaussian kernels follow the same premise of increasing the potential as samples get closer to each other, but instead of using real world distances, using image pixel distances. The kernels had to be carefully calibrated the proved similar results, however because of their non-linear relationship the do introduce a bias which shifts with distance from the camera.

4 Learning

Learning was done using unsupervised data, recorded in on an Xbox Kinect sensor. Data sets included both in-door and out-door videos, and featured both rigid and articulated object. Specifically a panning around several statues and recordings of my cat.

Because of the size of the data set and a lack of personal resources all data was unlabeled. The lack of ground truth made testing and evaluation very difficult to quantify. As such, I can only provide qualitative results as this time.

5 Results



The segmentation results seen above were typical of the segmentations achieved using the mixture of gaussians unary classifier. Better, supervised classifiers provide much more accurate segmentation, as most of the errors were in fact introduced by the mixture of gaussians unary classifier.

References

[1] P. Krahenbuhl and V. Koltun. Efficient Inference in Fully Connected CRFs with Gaussian Edge Potentials, *NIPS* (2011)

[2] A. Adams J. Baek and A. Davis. Fast High-Dimensional Filtering Using the Permutohedral Lattice, *Eurographics* (2010)