GeoInformatica 10:1, 37–53, 2006

Detecting Roads in Stabilized Video with the Spatio-Temporal Structure Tensor

ROBERT PLESS
Department of Computer Science and Engineering, Washington University, One Brookings Dr., St. Louis, MO, USA
E-mail: pless@cs.wustl.edu

Received 22 March 2005; Accepted 29 June 2005

Abstract

Video provides strong cues for automatic road extraction that are not available in static aerial images. In video from a static camera, or stabilized (or geo-referenced) aerial video data, motion patterns within a scene enable function attribution of scene regions. A "road," for example, may be defined as a path of consistent motion—a definition which is valid in a large and diverse set of environments. The spatio-temporal structure tensor field is an ideal representation of the image derivative distribution at each pixel because it can be updated in real time as video is acquired. An eigen-decomposition of the structure tensor encodes both the local scene motion and the variability in the motion. Additionally, the structure tensor field can be factored into motion components, allowing explicit determination of traffic patterns in intersections. Example results of a real time system are shown for an urban scene with both well-traveled and infrequently traveled roads, indicating that both can be discovered simultaneously. The method is ideal in urban traffic scenes, which are the most difficult to analyze using static imagery.

Key words: statistical video processing, stabilized video, aerial video

1. Introduction

Automatically populating databases with current information about road networks is important in the automatic acquisition and update of geographic information systems (GIS). Both civic planning and tactical response to emergency situations require such data to reflect current conditions. The extraction of roads from image data has led to significant scientific inquiry within the Computer Vision community developing tools for scale-invariant detection of features amidst significant and highly varied background clutter. The complexity of this problem requires image analysis systems to include significant semantic modeling, allowing context based reasoning to be used in image areas with ambiguous image data. Hinz states:

> *[It is clear that] detailed semantic modeling, contextual reasoning, and self diagnosis … must be integral parts of an extraction system to attain reasonably good results over a variety of scenes* [9]

While this assertion may be valid for extracting roads from single images, we argue here that the ambiguities may be mitigated in the analysis of video data from a scene.

Video data is particularly beneficial for urban scenes, where roads tend to be more difficult to identify from a single image but there is a high traffic volume and therefore consistent motion cues.

Historically, several problems have limited the use of video data in photogrammetry—the relatively low resolution of video and the massive and highly redundant form of the data set. These problems have been ameliorated with the wider availability of megapixel video cameras and algorithmic advances that allow real time stabilization (registering a video to an internally consistent coordinate system or geo-registration) and anomaly detection as tools for extracting efficient representations of the data. For the remainder of this paper we will assume that the video has been stabilized, so that motion within the video is caused by (1) objects moving in the scene, (2) the background motion of fixed objects in the scene (trees, water motion), or, in the case of aerial video, (3) residual (unstabilized) motions of static objects that are significantly above the ground plane and thus not stabilized.

This work is inspired by recent work in video surveillance—anomaly detection algorithms that are effective at modeling consistent background motions (eg., trees waving in the wind, water waves, or consistent traffic patterns) in order to trigger an alarm or to save data when an unusual event occurs or an object moves through the scene in an unusual manner [15]. The construction of these models, which are intended to capture the typical behavior of a scene, turns out to be an ideal pre-processing and video-data summarization step in terms of identifying roads in a scene. This approach is compelling for a number of reasons:

- The anomaly detection method is based on capturing the joint distribution of spatio-temporal image derivatives instead of continuously tracking objects and modeling typical trajectories. Therefore the data can be generated by many short time sequences (at least pairs of images) and does not require continuous imaging of the same area. This gives flexibility in the data capturing process and allows road extraction data analysis to be piggybacked on data captured for other purposes (such as aerial surveillance).
- The processing of the video data gives, for each pixel or small image region, the best fitting motion direction, and a measure of how consistent the image derivatives are with a single motion direction. This serves as a pre-processing step that may provide features to support many annotation tasks and can be integrated with tools such as snakes, condensation, and particle filtering.
- It is also possible to automatically discover patterns of motion within the scene that indicate, for example, different flow patterns of traffic through a traffic intersection.
- This method is effective at seeding static image analysis methods. Detecting roads based on consistent motion patterns is highly effective, but only for roads with visible traffic. The parameters of the roads detected in this manner (the image size, color, typical curvature, etc) can be used to seed image based methods with parameters specific to the given data set, in order to detect the remaining roads in the scene.
- Finally, the representation may be augmented to capture additional information depending on the length of time a scene is observed, including the volume or frequency of travel along the road and the distribution of vehicle speeds.

The following section attempts to place this work in the context of recent approaches to road extraction. Section 3 introduces the real-time approach to spatio-temporal image processing and techniques to maintain a representation of the motion distributions at each pixel. Section 4 gives implementation details for using this representation for road detection and a demonstration in an urban scene. Section 5 illustrates a natural extension to the technique to discover motion patterns which allows classification of traffic intersections, and a short discussion follows.

2. Background

Many of the systems for road extraction can be categorized in terms of their (1) front end sensors, (2) initial data filtering and analysis at the pixel or local level, and (3) methods to define extended paths on the basis of initial image data. Here we present a sparse survey of recent literature on road extraction methods as a means of putting our proposed approach into context. Although our approach is defined explicitly in subsequent sections, for comparison purposes, it would be categorized in this framework as using (1) aerial video and (2) extended spatiotemporal filtering. We are explicitly agnostic about the third component, and emphasize that the front end processing we propose can be used within existing multiresolution, active testing, or snake based models for detecting roads.

Geman and Jedanyk discuss road extraction from satellite imagery [7]. They argue that immediately classifying pixels as “road” or “background” is infeasible because the local region surrounding a road pixel and a background pixel may appear identical (even in multi-spectral LandSat imagery). Thus they propose a particular brightness invariant local operator and use an active testing approach that follows the road appearance and path by minimizing an entropy measure. Additional methods improve road detection by integrating larger local windows of image appearance. Exemplars for this approach advocate multiscale analysis for the extraction of road network from multi-spectral imagery [22], and using snakes as a method of finding long regions that are straight or curve slowly [12], or the combination of multi-spectral imaging and a selforganizing road map which preferentially converge to smooth road paths [4]. A stochastic representation of road appearance based on snakes has been proposed to take advantage of multiple images for shape optimization and change detection [1].

Focusing more on the data analysis at the pixel level, Porikli proposes a set of line-filters that measures both how likely a particular pixel is a road, as well as the direction of the possible road at that pixel. As the algorithm progresses, the ends of currently detected roads can be extended to areas that have a very low likelihood of being a road, as long as they have the correct orientation [16]. Related methods define the road as a probabilistic contour, and use color and gradient information to extend contours across occlusions or shadows [2].

Synthetic aperture radar (SAR) has been considered as a front end sensor to simplify the process of road extraction. Tupin considers the problem of road extraction in urban areas, and proposes a 2-step algorithm that extracts line features from the speckle radar image and subsequently uses a Markov random field to impose contextual knowledge to cluster the detected segments into roads [19]. Wessel argues

that road extraction from SAR imagery is effective for highways where there are no scattering objects (signs, or bridges) that interfere with the road, but are ineffective in industrial areas (which tend to always have scattering objects) or for secondary roads (which tend to have insufficient signal return).

Returning to aerial imagery, the papers most closely related to this approach concentrate on road extraction in urban environments. Hinz points out that most work focuses on the easier problem of rural road extraction, and existing work on urban scenes make assumptions about the grid structure of many city streets [5], or combines height models and high-resolution imagery to extract streets through residential areas [17]. To be effective in more general situations, a system is proposed that incorporates a great deal of detailed knowledge about roads and their context, uses explicit formulated scale-dependent models of road appearance, and continually performs hypothesis testing to ensure that the local context information is appropriate [9].

Finally, from the computer vision community, work on traffic monitoring using a "Forest of sensors," is effective at creating trajectories of objects tracked through an environment [8], [18]. While this could form the basis for an approach similar to ours, they do not consider the problem of road extraction, and their method requires continuous long term surveillance to build trajectories, instead of capturing and integrating short term motion cues.

A technical issue directly related to our approach (otherwise independent of road extraction) is that we require the input video sequence to be stabilized, so that collecting statistics of spatio-temporal filter responses over many frames at a single pixel gives motion information about the same scene point. Numerous algorithms for this process have been proposed using either the tracking of feature points [23], or based directly on spatiotemporal filter responses [3], [14]. We adapt the method used in [14] which involves, for each frame, computing spatio-temporal filters at a sparse set of image points, and solving for a general linear transformation (the image warping homography) that minimizes the change from the previous frame. The sequence of the warped images becomes the stabilized video used as input to the algorithm described below. Alternatively, video data that is tagged with very accurate knowledge of the 3D position and orientation of the camera in each frame may permit warping and stabilization without additional image processing.

In summary, to our knowledge, no one has directly considered the question of using aerial video imagery in the detection of roads. Recent advances in computational power and algorithmic maturity make the use of video data feasible. Using video imagery to define roads based upon motion patterns is most effective in urban environments—a domain that remains particularly challenging for both image and SAR based analysis.

3. Motion features: Spatio-temporal structure tensor

The approach is based upon spatiotemporal image analysis. This approach explicitly avoids finding or tracking image features. Instead, the video is considered to be a 3D function I(x,y,t), defining the image intensity as it varies in space (across the image)

and time. The fundamental atoms of the image processing are the value of this function and the response to spatio-temporal filters (such as derivative filters), measured at each pixel in each frame. Unlike interest points or features, these measurements are defined at every pixel in every frame of the video sequence. Appropriately designed filters may give robust measurements to form a basis for further processing. Optimality criteria and algorithms for creating derivative and blurring filters of a particular size have been developed by [6], and lead to significantly better results than estimating derivatives by applying Sobel filters to raw images. For these reasons, spatio-temporal image processing provides an ideal first step for streaming video processing applications.

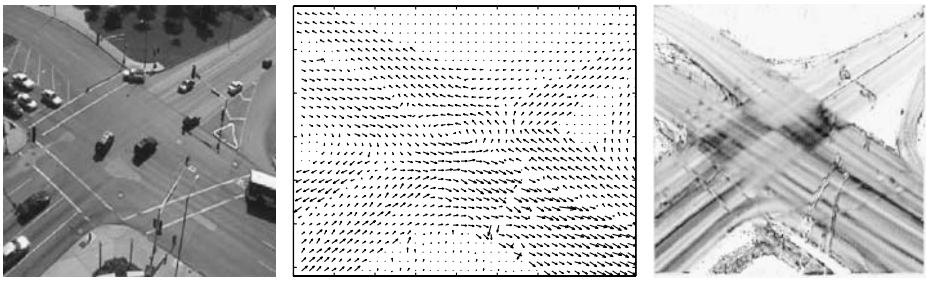
Space and time derivative filters are particularly meaningful in the context of analyzing motion on the image. Considering a specific pixel and time (x,y,t), we can define $I_x(x, y, t)$ to be the derivative of the image intensity as you move in the x-direction of the image. $I_y(x, y, t)$, and $I_t(x, y, t)$ are defined similarly. At a given pixel at a given time, and the optic flow constraint equation gives a relationship between I_x, I_y, and I_t, and the optic flow, (the 2d motion at that part of the image) [10]:

$$I_x u + I_y v + I_t = 0.$$

This classical equation in computer vision holds true for smoothly varying images, when the motion (the magnitude of the $\langle u, v \rangle$ vector) is relatively small, and the only reason that the intensity at a pixel changes is because of local motion in the image. This gives just one equation with two unknowns (u, v) so it is not possible to directly solve for the optic flow. Many optic flow algorithms therefore assume that the optic flow is constant over a small region of the image, and use the (I_x, I_y, I_t) values from neighboring pixels to provide additional constraints. This assumption does not hold true at the boundaries of objects, leading to consistent errors in the optic flow solution.

The advantage of using *stabilized* video for scenes with consistent motion patterns is that instead of combining data from spatially extended region of the image, we can instead combine equations through time. This allows one to compute the optic flow at a single pixel location without any spatial smoothing. Figure 1 shows one frame of a video sequence of a traffic intersection, and the flow field that best fits the data for each pixel over time. The key to this method is that the distribution of the spatio-temporal intensity derivatives observed at a pixel (simply the distribution, and not, for instance the time sequence) encodes several important parameters of the underlying variation at each pixel.

In this section we introduce our approach to representing the distribution of image derivatives captured at each pixel. We use the spatio-temporal structure tensor field (the covariance matrix of the space and time image derivatives accumulated through time at each pixel), which has dual benefits: first, the parameters are efficient to update and maintain within a real-time application, and second, the set of parameters for the entire image efficiently summarizes and encodes features of interest to GIS applications. Furthermore, the structure tensor field admits a natural method of parsing the scene into multiple motion components, which are common, for example, at intersections where traffic lights enforce varied traffic patterns.



(a) Video (b) Estimated Flow (c) Confidence

Figure 1. Flow estimated from 3-D structure tensor for 10 minutes of video of an intersection. Multiple motions at the middle of the intersection cause a circular pattern in the estimated flow field. The cofidence decreases in the parts of the middle of the intersection where the angle between the two major motions is larger. Inside the intersection, the two "quadrants" which see motion with directions differing by more than 90° have lower confidence that the other two quadrants.

3.1. Representing the distribution of derivatives

In this section we introduce the spatio-temporal structure tensor. We denote the spatio-temporal image derivatives by the column vector ∇I, so that:

$$\nabla I(\vec{p},t) = \left(I_x(\vec{p},t), I_y(\vec{p},t), I_t(\vec{p},t)\right)^\top,$$

are the spatio-temporal derivatives of the image intensity $I(\vec{p},t)$ at pixel $\vec{p}$ and time t. The covariance matrix of a particular spatial temporal measurement ∇I can be written as $\nabla I \nabla I^\top$. We define the structure tensor Σ at pixel $\vec{p}$ to be the average of the covariance matrices of all derivative measurements observed at that pixel:

$$\Sigma(\vec{p}) = \frac{1}{f}\sum_{t=1}^{f} \nabla I(\vec{p},t)\nabla I(\vec{p},t) = \frac{1}{f}\begin{pmatrix} \sum_1^f I_x^2 & \sum_1^f I_x I_y & \sum_1^f I_x I_t \\ \sum_1^f I_x I_y & \sum_1^f I_y^2 & \sum_1^f I_y I_t \\ \sum_1^f I_x I_t & \sum_1^f I_y I_t & \sum_1^f I_t^2 \end{pmatrix}$$

where f is the number of frames in the sequence and we omit $\vec{p}, t$, in the matrix shown above and hereafter for the sake of clarity. Except as described in Section 3.1. Here we model these distributions independently at each pixel, although we recognize that in real scenes there may be significant correlations and in Section 5 we explicitly reason about these correlations. To focus on scene motion, the measurements are filtered based on their I_t value, only considering measurements $\nabla I(\vec{p},t)$ that come from variation in the scene. We choose to use a simple threshold, and incorporate into the model all measurements for which $|I_t| > 2$ (a change of at least two out of 256 gray levels between consecutive frames).

In real-time applications, batch computation over the entire sequence is not feasible and the structure tensor must be estimated online. Assuming the distribution is stationary, the structure tensor Σ_t at time t can be estimated exactly as a weighted combination of the structure tensor at time $t-1$, and the covariance matrix of the current measurements, $\nabla I \nabla I^\top$:

$$\Sigma_t = \frac{(n-1)}{n}\Sigma_{t-1} + \frac{1}{n}\nabla I \nabla I^\top.$$

The structure tensor has a number of interesting properties that are exposed through computation of its eigenvalues and eigenvectors. In particular, suppose that Σ has eigenvalues (sorted from largest to smallest) λ_1, λ_2, λ_3, and corresponding eigenvectors $\left(\overrightarrow{v}_1, \overrightarrow{v}_2, \overrightarrow{v}_3\right)$. The following properties hold:

- The vector $\overrightarrow{v}_3$ is a homogeneous representation of the total least squares solution [11, 20, 21], for the optic flow. The 2-d flow vector (f_x, f_y) can be written:

$$\left(f_x, f_y\right) = \left(\frac{v_3(1)}{v_3(3)}, \frac{v_3(2)}{v_3(3)}\right)$$

- If, for all the data at that pixel, the set of image intensity derivatives exactly fits some particular optic flow, then λ_3 is zero.
- If, for all the data at that pixel, the image gradient is in exactly the same direction, then λ_2 is zero. (This is the manifestation of the classical aperture problem).
- The consistency value $C = 1 - \frac{\lambda_3}{\lambda_2}$, is a an indicator of how consistent the image gradients are with the best fitting optic flow, with 1 indicating perfect fit, and 0 indicating that many measurements do not fit this optic flow.
- The specificity value $\mathbf{S} = \frac{\lambda_2}{\lambda_1}$, varies from 0 to 1, and is an indicator of how well specified the optic flow vector is. When this number is close to 0, the image derivative data could fit a family of optic flow vectors with relatively low error, when this ratio is closer to 1, then the best fitting optic flow is better localized.

These are properties of the spatiotemporal structure tensor accumulated through time at each pixel. Considering the entire image, we can use these properties to define useful features at each pixel (x, y). Our claim is that these variables are an effective summary of information contained in a video sequence, and that the analysis of the following scalar, vector, and tensor fields is an effective method for extracting road features from stabilized video:

- $\Sigma(x, y)$, the tensor field consisting of the structure tensor at each pixel,
- $\langle f_x(x, y), f_y(x, y)\rangle$, the best fitting optic flow field,
- $s(x, y)$, the specificity of the optic flow solution, and
- $c(x, y)$, the consistency of the optic flow solution.

The next section gives implementation details that indicate how to use these variables to detect roads.

4. Implementation details

A formal evaluation is limited by the lack of standardized test data and the absence of other algorithms which compute GIS features from stabilized aerial video. To address this problem in the future, the code and data sets presented in the following section are publicly available.[1] Here we completely define the approach taken for an example data set, indicate the results, point out limitations and indicate areas of potential future research.

4.1. Methods

We have found that the method is quite robust to various implementation choices, but for concreteness we describe here the exact choices used in the results. Consecutive pairs of images from the video sequence are decompressed to create 2D arrays of intensity values. The image is convolved with a discrete 11 by 11 filter approximating a Gaussian with standard deviation of three pixels to create a blurred image. The I_x and I_y values are computed with appropriately oriented Sobel filters convolved with the blurred image. The I_t value is estimated as the difference between pixel values in consecutive frames. This (I_x, I_y, I_t) measurement is maintained for every pixel in the image, but is ignored at pixels whose distance from the boundary is less than 6 pixels, as the results of the convolution filters at these points depends upon assumptions about pixel values outside the image. At each pixel, a 3 $\times$ 3 covariance matrix is maintained by storing 7 parameters, (ΣI_x^2, ΣI_y^2, ΣI_t^2, $\Sigma I_x I_y$, $\Sigma I_x I_t$, $\Sigma I_y I_t$, n). To isolate the effects of image motion from intensity gradients that exist in the static image, these sums and the value of n are only updated when $|I_t| > 2$. The number n records the number of measurements that have been used in each of the sums. We emphasize that the above sums are taken through time, and these parameters are recorded separately for each pixel, so some pixels may have more measurements that define their covariance matrix than others.

4.2. Results

The original image and several of the results of the preprocessing methods described above are shown for a 451 frame stabilized aerial video. This video is taken at approximately 3 frames per second. The video was shot from an aerial platform and georegistered. Two frames of the geo-registered video are shown at the top of Figure 2, the black areas at the bottom corners arise because these images are warped to the coordinate system of the reference frame and these areas were outside the image. For each pixel, a score is calculated to measure how likely that pixel is to come from a road. This score function is:

$$\mathrm{SC}\Sigma I_t^2,$$

which is the intensity variance at that pixel, modulated by the previously defined scores that measure how well the optic flow solution fits the observed data (C) and

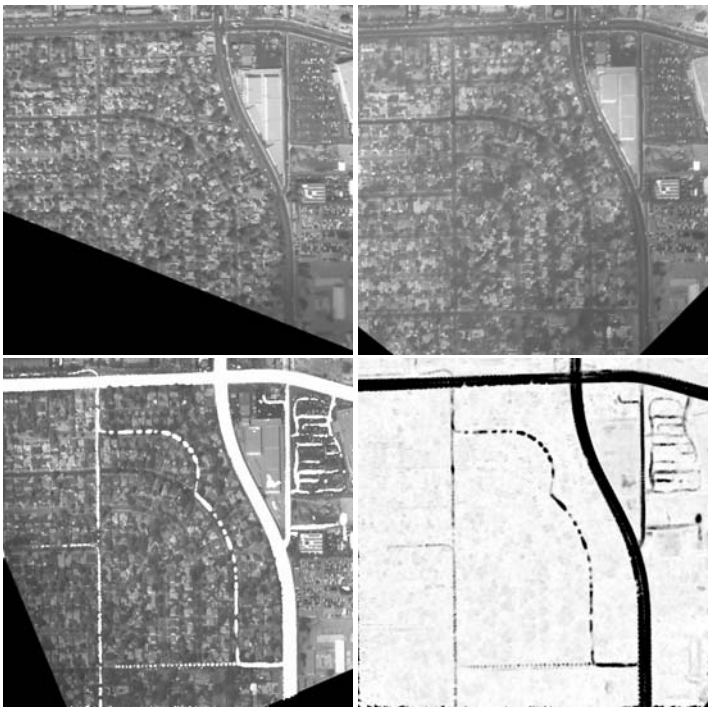


Figure 2. The top row shows frames 1 and 250 of a 451 frame stabilized aerial video (approximately 2:30 minutes long, 3 frames per second). The black in the corners are areas in this geo-registered frame that are not captured in these images, these areas are in view for the majority of the sequence. The bottom right shows the amount of image variation modulated by the motion consistency—a measure of how much of the image variation is caused by consistent motion (as would be the case for a road). These scores are overlayed onto an original image in the bottom left.

how unique that solution is (S). This score is thresholded (threshold value set by hand), and overlayed on top of the original image in the bottom left of Figure 2.

Road are detected in image regions where cars are visible moving during the video input, including regions where image based detection systems would fail, such as the upper of the two curved roads in the middle of the image. This justifies the earlier assertion that this system is ideal in urban areas, where the image based cues are less clear but where the frequency of traffic is sufficient that motion cues are often available.

However, the motion cues provide more information than simply a measure of whether the pixel lies on a road. The best fitting solution for the optic flow also gives the direction of motion at each pixel. The components of the motion vectors are shown as the top row of Figure 3. There is significant noise in this motion field because of substantial image noise and the fact that for some roads the data included few moving vehicles. A longer image sequence would provide more data and make flow fields that are well constrained and largely consistent. The method would continue to fail in regions that contain multiple different motion directions or where the optic flow constraint equations fail. The next section introduces a post-processing step to make the flow field analysis feasible with shorter video sequences.

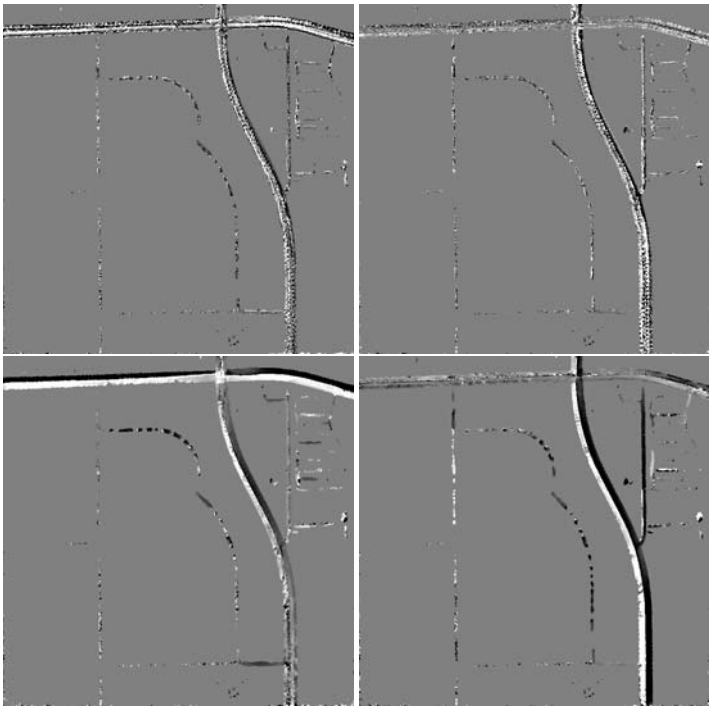


Figure 3. The top row show the x and y components of the best fitting optic flow vectors for the pixels designated as roads in Figure 2. The flow fields are poorly defined, in part because of noisy data, and in part because there were few cars that move along some roads. These (poor) flow estimates were used to define the directional blurring filters that combine the image intensity measurements from nearby pixels (forward and backwards in the direction of motion). Using the covariance matrix data from other locations along the motion direction gives significantly better optic flow measurements (bottom row). In these images, black is negative and white is positive, relative to the origin in the top left corner of the image.

4.3. Post-processing

In the presence of noise within the flow fields, it is necessary to combine information between nearby pixels. Typically, combining information between pixels leads to blurring of the image and a loss of fidelity of the image features. However, the flow field extracted gives a best fitting direction of travel at each pixel. We use this as a direction in which we can combine data without blurring features—that is, we use the noisy estimate of the motion as a cue to help combine data along the roads, rather than across roads.

In particular, we adapt previous work on smoothing flow fields, called motion oriented averaging [13], but instead of averaging optic flow estimates, we compute the weighted average of nearby covariance matrices. This process first defines a local weighting function at each pixel by computing a normalized direction vector $\langle d_x, d_y \rangle$. A spatial covariance matrix M is created with this vector as the principle Eigenvector, in order to define a Gaussian weighting function w which is oriented so that larger

values lie along the direction of motion. More specifically, we follow the following algorithm:

Algorithm:

1. Compute a normalized vector in the direction of motion

$$\langle d_x, d_y \rangle = \frac{\langle f_x(x,y), f_y(x,y) \rangle}{\sqrt{f_x^2(x,y) + f_y^2(x,y)}}$$

2. Define weights with higher weight along orientation of motion

$$T = \begin{bmatrix} 30d_x & 30d_y \\ -3d_y & 3d_x \end{bmatrix}$$

$$M = T^\top T$$

$$w(a,b) = e^{-\langle a,b \rangle^\top M^{-1} \langle a,b \rangle}$$

3. Compute weighted average of nearby structure tensors

$$\widehat{\Sigma}(x,y) = \sum_{a=-5\ldots5} \sum_{b=-5\ldots5} w(a,b)\Sigma(x+a, b+y)$$

This process serves to reduce noise in the estimates of the structure tensor without average across different directions of traffic. The bottom row of Figure 3 depicts the flow field components derived from the smoothed covariance matrix field (the third eigenvector of the matrices $\widehat{\Sigma}(x,y)$). These fields are significantly cleaner than the flow field computed from the covariance matrices without smoothing and show clearly defined directions of travel for all roads which had motion. Furthermore, for large roads with significant motion cues, there is a clearly defined separation between different directions of travel. The motion patterns at the intersections, however, cannot be cleanly described with a single model. The following section works to consider the analysis of these regions with multiple motion patterns.

4.4. Analysis of results and algorithmic efficiency

While we believe that the analysis of stabilized aerial video will be increasingly important in the future, there is currently a paucity of publicly available sample imagery for testing of this approach. However, in this section we attempt to identify the limitations of our approach with respect to various parameters of the input video.

This approach uses motion to detect road locations. Therefore the video duration must be long enough so that some vehicle is seen moving along each road in the scene. Furthermore, the distance a car travels per frame must be less than half the size of the filters (described at the beginning of Section 3) used to estimate the spatio-temporal image derivatives, otherwise the filter responses are not correlated with object motion. Additionally, apparent scene motion not due to vehicles is mis-categorized as roads (for example, residual motion for poorly registered aerial video or specular reflections off of drainage canals often lead to apparent motion).

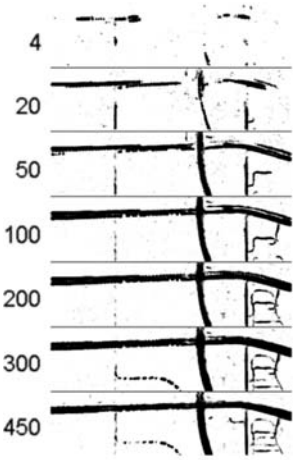


Figure 4. Isolating the top of the video sequence, we show the score function from Equation 1, as computed when using 4, 20, 50, 100, 200, 300, or 450 consecutive frames of an input sequence taken at 2 frames per second. Notice the dotted appearance of the top horizontal road after just a few frames, and the decrease in the score function for the left-most vertical road.

Our implementation of the algorithm as described through Section 4, continuously updating the structure tensor and computing the scoring function (Equation 1), runs in real time (30 fps for a 512 by 512 image size) on commodity hardware (a 2.0 GHz Pentium PC). The post-processing step of Section 4.3 takes several seconds but is only done once per video, and the running time depends only on the size of the image, not the length of the video. This efficiency implies that there is no algorithmic limit to the maximum length of the input video.

Figure 4 shows the effect of using progressively more frames of video. For densely travelled urban scenes, even 4 frames of video is sufficient to detect consistent motions patterns along some roads. Also notice that the score function may decrease over time. In the figure, a car passes along vertical road on the left side of the figure within the first 20 frames, but over the subsequent hundreds of frames, other image variation (largely due to clouds) decreases the road detection score because the image derivative distribution is no longer consistent with just one motion direction. The initial dotted appearance of the top road is due to individual vehicles that move too far between frames to see continuous motion—as more vehicles pass through the same region of road this is filled in. The dotted appearance of the curved road in the bottom middle of the scene is due to partial occlusion from trees that obscure the vehicles at some locations.

5. Motion patterns: Mixtures of structure tensor fields

The last section labels pixels based on the local distribution of spatio-temporal intensity derivatives. In this section we consider the case that there are multiple global

motion patterns in a scene (such as in a traffic intersection). Factoring the scene into coherent motion patterns allows more accurate motion estimates in areas such as traffic intersections. This can be achieved by considering the image derivatives from all pixels in a given frame as a single measurement vector and using adaptive mixture models to find a small number of models that account for all the data [24].

The structure tensor, as described in the last section, is the covariance matrix, Σ, of the measurements, ∇I, at each pixel. Assuming independence between pixels (a false assumption, but one which makes the mathematics tractable), then the structure tensor field may be considered as a single joint Gaussian, $\mathcal{N}_{global}$, over the entire image. If Σ_i is the structure tensor at the i-th pixel, the covariance matrix of the global distribution is the block-diagonal matrix:

$$\widetilde{\Sigma}_{global} = \begin{pmatrix} \Sigma_1 & & & 0 \\ & \Sigma_2 & & \\ & & \ddots & \\ 0 & & & \Sigma_p \end{pmatrix}$$

This defines a distribution over the set of all image derivative measurements in the image at a particular frame. This measurement can be written as $\widetilde{\nabla} I$, the concatenation of the gradient vector at each individual pixel: $\widetilde{\nabla} I = \left(I_x^{(1)}, I_y^{(1)}, I_t^{(1)}, I_x^{(2)}, \ldots\right)$. If the motions visible in the scene come from multiple motion patterns, then the image derivative measurements, $\widetilde{\nabla} I$, do not derive from just one global model, $\widetilde{\Sigma}$, but rather from several. We solve for these several motions models using the adaptive mixtures framework, an online method which maintains several global motion models and updates the models with new data by an amount proportional to the likelihood that the new data comes from each model[2]. Given a model, $\mathcal{N}_{global}$, then the likelihood of observing global image derivatives, $\widetilde{\nabla} I$, is:

$$P\left(\widetilde{\nabla} I \middle| \mathcal{N}_{global}\right) = k \exp\left(-\frac{1}{2} \widetilde{\nabla} I^{\top} \widetilde{\Sigma}_{global}^{-1} \widetilde{\nabla} I\right)$$

where k is a normalizing constant. Because $\widetilde{\Sigma}$ is block diagonal, this can be rewritten as:

$$P\left(\widetilde{\nabla} I \middle| \mathcal{N}_{global}\right) = \prod_i P(\nabla I_i | \mathcal{N}_i(0, \Sigma_i)),$$

which allows this computation to be efficient. The adaptive mixture model defines the distribution of measurements as the sum of a collection of Gaussian distributions:

$$w_1 \mathcal{N}_1\left(0, \widetilde{\Sigma}_1\right) + \cdots + w_M \mathcal{N}_M(0, \Sigma_M)$$

where M is the number of models. The adaptive mixture model update equations (as used, for example, in Stauffer and Grimson [18]) allow this model to be automatically acquired. Here it is applied to a very high-dimensional distribution (with a special

block diagonal structure). Updating the mixture model online requires first calculating the likelihoods:

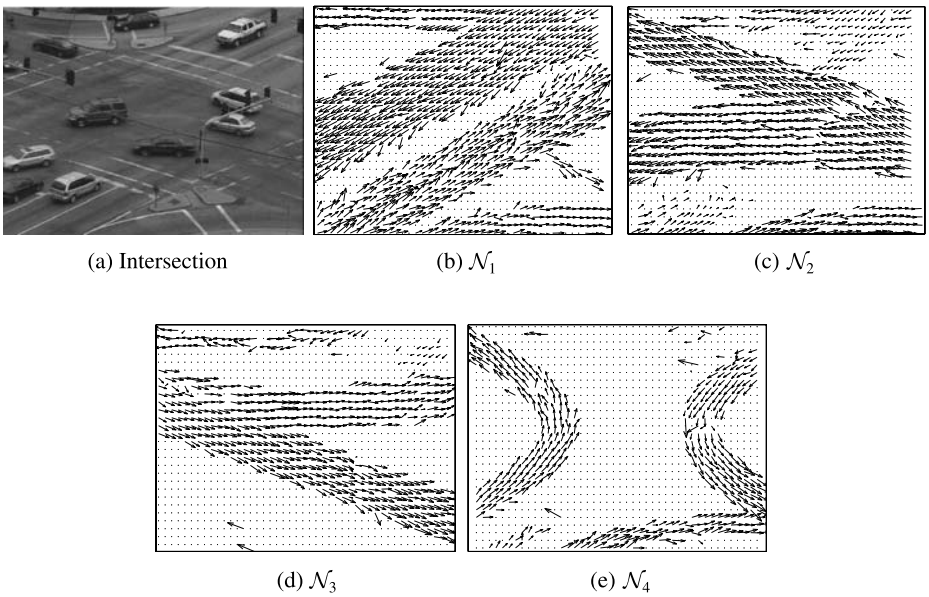
$$P\left(\mathcal{N}_i \middle| \widetilde{\nabla} I\right) = \frac{w_i P\left(\widetilde{\nabla} I | \mathcal{N}_i\right)}{\sum_{j=1}^{M} w_j P\left(\widetilde{\nabla} I | \mathcal{N}_j\right)},$$

and then updating each fields as:

$$\widetilde{\Sigma}_{i,t} = (1 - \beta_i)\widetilde{\Sigma}_{i,t-1} + \beta_i \widetilde{\nabla} I \, \widetilde{\nabla} I^{\top}$$

with a weighting factor $\beta_i = P\left(\mathcal{N}_i | \widetilde{\nabla} I\right)$, which is proportional to the probability that $\mathcal{N}_i$ the correct model. The complete update of the adaptive mixture model requires that the weights of the components be adjusted. The weights w_i can be updated as $w_{i,t} = (1 - \beta_i)w_{i,t-1} + \beta_i$. It is important to note that each component of the mixture model retains the block diagonal, so that the constraints on the derivative measurements at each pixel are independent, except in their contribution to determining the likelihoods that each model fits the global data.

Figure 5 shows the results of a real time system applying this adaptive mixture model to a 10 minute video scene dominated by an intersection. The four components



(a) Intersection (b) $\mathcal{N}_1$ (c) $\mathcal{N}_2$

(d) $\mathcal{N}_3$ (e) $\mathcal{N}_4$

Figure 5. Coherent motion fields [24] serve to automatically parse complicated intersection flow fields into their components traffic patterns. Four mixture components are estimated by applying adaptive mixture models to the distribution of global image derivative measurements. This serves as a cue that may support high fidelity annotation of types of intersections.

of the mixture model define distributions over the global image derivative measurements. Decomposing each mixture model into its block diagonal components, and solving for the best fitting optic flow of each component, gives a visualization of the component flow fields within this scene. Notice that these flow fields indicate that this intersection has one traffic directions whose light cycle is "straight in both directions then a left turn only" $(\mathcal{N}_1, \mathcal{N}_4)$, and the other direction of travel is "go straight and turn left then (in the reverse direction) go straight and turn left" $(\mathcal{N}_2, \mathcal{N}_3)$. This decomposition derives from video data alone, and does not require any synchronization with the traffic light.

6. Discussion

This paper has presented a robust algorithm for the preprocessing of stabilized video, marking pixels whose intensity variation is consistent with motions along a road. Furthermore, the direction of travel can be accurately identified, and varied patterns of motion may possible help characterize intersections. The motion cues are stronger for regions with significant traffic, and therefore most useful in urban settings. These settings are very challenging for current approaches based on the analysis of static images because of the typical complexity of the background.

6.1. Future work

Many avenues are open for future research, including joining motion based algorithms with more standard appearance based algorithms, as well as fitting the motion cues into larger systems that output symbolic representations of roads rather then pixel scores. Concrete future directions include:

- Integrate appearance or motion models specific to intersections, parking lots, and other features of interest.
- Use the motion based analysis of motion to bootstrap appearance based road modeling. For example, in Figure 2, not all roads in the scene are discovered because some roads did not have cars pass along them during the input video. However, statistical appearance models of the roads identified through motion cues would give a scene-specific road appearance model to find those roads that were missed.
- Adapt snake based road following algorithms (such as Laptev et al. [12]) to incorporate the flow field direction in the energy function minimized by the snake.
- Design algorithms that close the loop between region interpretation and anomaly detection and detect unusual events such as cars stopping in unusual places.

Acknowledgment

This work was supported under NSF grant IIS-0413291.

Notes

1. http://www.cse.wustl.edu/~pless/videoGIS.html
2. For clarity, this presentation ignores special case issues in initializing the mixture model and identifying data that comes from none of the current models. We use exactly the approaches defined (for another application) in Stauffer and Grimson [18].

References

1. P. Agouris, A. Stefanidis, and S. Gyftakis. "Differential snakes for change detection in road segments," *Photogrammetric Engineering & Remote Sensing*, Vol. 67(12):1391–1399, 2001.
2. M. Bicego, S. Dalfini, G. Vernazza, and V. Murino. "Automatic road extraction from aerial images by probabilistic contour tracking," in *Proceedings of IEEE International Conference on Image Processing (ICIP)*, vol. III, pp. 585–588, 2003.
3. X.-T. Dai, L. Lu, and G. Hager. "Realtime video mosaicing with adaptive parameterized warping," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, volume (Demo Program), 2001.
4. P. Doucette, P. Agouris, A. Stefanidis, and M. Musavi. "Self-organized clustering for road extraction in classified imagery," *ISPRS Journal of Photogrammetry and Remote Sensing*, Vol. 55(56):347–358, 2001.
5. A. Faber and W. Forstner. "Detection of dominant orthogonal structures in small scale imagery," *International Archives of Photogrammetry and Remote Sensing*, 33(Part B3/1):274–281, 2000.
6. H. Farid and E.P. Simoncelli. "Optimally rotationequivariant directional derivative kernels," in *Computer Analysis of Images and Patterns (CAIP)*, pp. 207–214, 1997.
7. D. Geman, and B. Jedynak. "An active testing model for tracking roads in satellite images," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 18(1):1–14, 1996.
8. W.E.L. Grimson, C. Stauffer, R. Romano, and L. Lee. "Using adaptive tracking to classify and monitor activities in a site," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 22–27, 1998.
9. S. Hinz, and A. Baumgartner. "Automatic extraction of urban road networks from multiview aerial imagery." *ISPRS Journal of Photogrammetry and Remote Sensing*, 53:83–98, 2003.
10. B.K.P. Horn. "Robot Vision." McGraw Hill: New York, 1986.
11. S. Van Huffel and J. Vandewalle. "The Total Least Squares Problem: Computational Aspects and Analysis," Society for Industrial and Applied Mathematics, Philadelphia, 1991.
12. I. Laptev, H. Mayer, T. Lindeberg, W. Eckstein, C. Steger, and A. Baumgartner. "Automatic extraction of roads from aerial images based on scale space and snakes," *Machine Vision and Applications*, Vol. 12(1):23–31, 2000.
13. H.H. Nagel. "Extending the 'oriented smoothness constraint' into the temporal domain and the estimation of derivatives of optical flow," in *Proceedings of the First European Conference on Computer Vision*, Springer: Berlin Heidelberg New York, Inc., pp. 139–148, 1990.
14. R. Pless, T. Brodsky, and Y. Aloimonos. "Detecting independent motion: The statistics of temporal continuity," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 22(8):68–73, 2000.
15. R. Pless, J. Larson, S. Siebers, and B. Westover. "Evaluation of local models of dynamic backgrounds," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 73–78, 2003.
16. F.M. Porikli. "Road extraction by pointwise gaussian models," in *SPIE AeroSense Technologies and Systems for Defense and Security*, vol. 5093, pp. 758–764, 2003.
17. K. Price. "Urban street grid description and verification," in *IEEE Workshop on Applications of Computer Vision*, pp. 148–154, 2000.
18. C. Stauffer, and W.E.L. Grimson. "Adaptive background mixture models for realtime tracking," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 246–252, 1999.
19. F. Tupin, H. Maitre, J.F. Mangin, J.M. Nicolas, and E. Pechersky. "Detection of linear features in SAR images: Application to the road network extraction," *IEEE Transactions on Geoscience and Remote Sensing*, Vol. 36(2):434–453, Mar. 1998.

20. S. Wang, Y. Markandey, and A. Reid. “Total least squares fitting spatiotemporal derivatives to smooth optical flow fields,’’ in *Proc. of the SPIE: Signal and Data Processing of Small Targets*, vol. 1698, pp. 42–55. SPIE, 1992.
21. J. Weber, and J. Malik. “Robust computation of optical flow in a multiscale differential framework,’’ *International Journal of Computer Vision*, Vol. 14:67–81, 1995.
22. C. Wiedemann, C. Heipke, H. Mayer, and S. Hinz. “Automatic extraction and evaluation of road networks from moms-2p imagery,’’ *International Archives of Photogrammetry and Remote Sensing*, Vol. 32(3): 285–291, 1998.
23. L. Wixson. “Detecting salient motion by accumulating directionallyconsistent flow,’’ *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 22(8):774–780, 2000.
24. J. Wright, and R. Pless. “Analysis of persistent motion patterns using the 3d structure tensor,’’ in *Proceedings of the IEEE Workshop on Motion and Video Computing*, pp. 14–19, 2005.

Robert Pless is currently an Assistant Professor of Computer Science and Assistant Director of the Center for Security Technologies at Washington Univerisity in St. Louis. A native of Baltimore, Maryland, he graduated from Cornell University with a Bachelor of Science and Computer Science in 1994, and received a Ph.D. in Computer Science from the University of Maryland in 2000. His field of research is Computer Vision, with a concentration in extreme camera geometries, panoramic vision, surveillance, and manifold learning, and he served as chairman of the 2003 IEEE International workshop on Omni-directional Vision and Camera Networks (Omnivis ’03).