Background Modeling and Blob Detection for an Intelligent Tracking System

Motivation

Develop a background modeling algorithm that processes video files or in real time as well as a blob detection algorithm that locates objects of interest in each frame of footage. Both algorithms will be used in a larger system called the Intelligent Tracking System (ITS). The ITS is a multi-object tracking system that has multiple levels of functionality as well as practically unlimited capacity for expansion.

Applications

Background Detection:

• Traffic monitoring and automatic emergency dispatch.

• Background template comparison surveillance.

Blob Detection:

• Interior surveillance systems

• Backbone of fast and multi-object tracking systems

Results & Future Work

The background subtraction algorithm works perfectly at about 10 frames per second. In order for it to run faster it could be optimized further as well as run on a faster system. Blob detection is still being debugged and should be ready for testing soon.

After these steps are finished it will be plugged in for use in a tracking algorithm that would then be expanded to multiple objects

Program Flow

Background Modeling

Takes an input image and looks at each pixel and each color channel then manipulates Gaussian functions based off of received pixel data. Statistically builds and stores the most likely background. The number of Gaussian functions used allows for more adaptive background modeling (i.e. allows for motion in the background).

Input Image
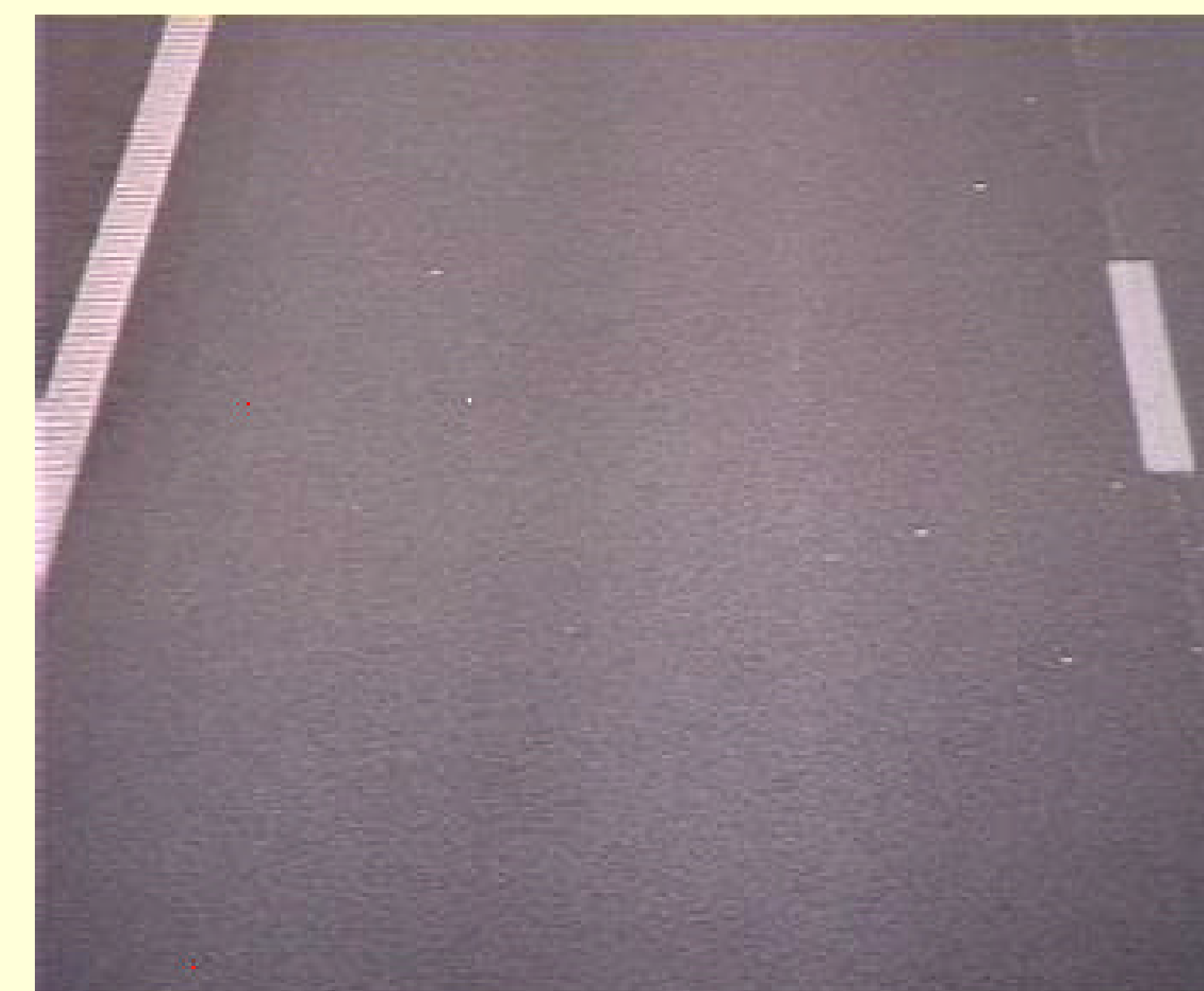
Output Image

Background Subtraction

Using the output from the background modeling algorithm, a simple threshold is set that is used to then erase the background from the picture leaving just the foreground (or moving objects)
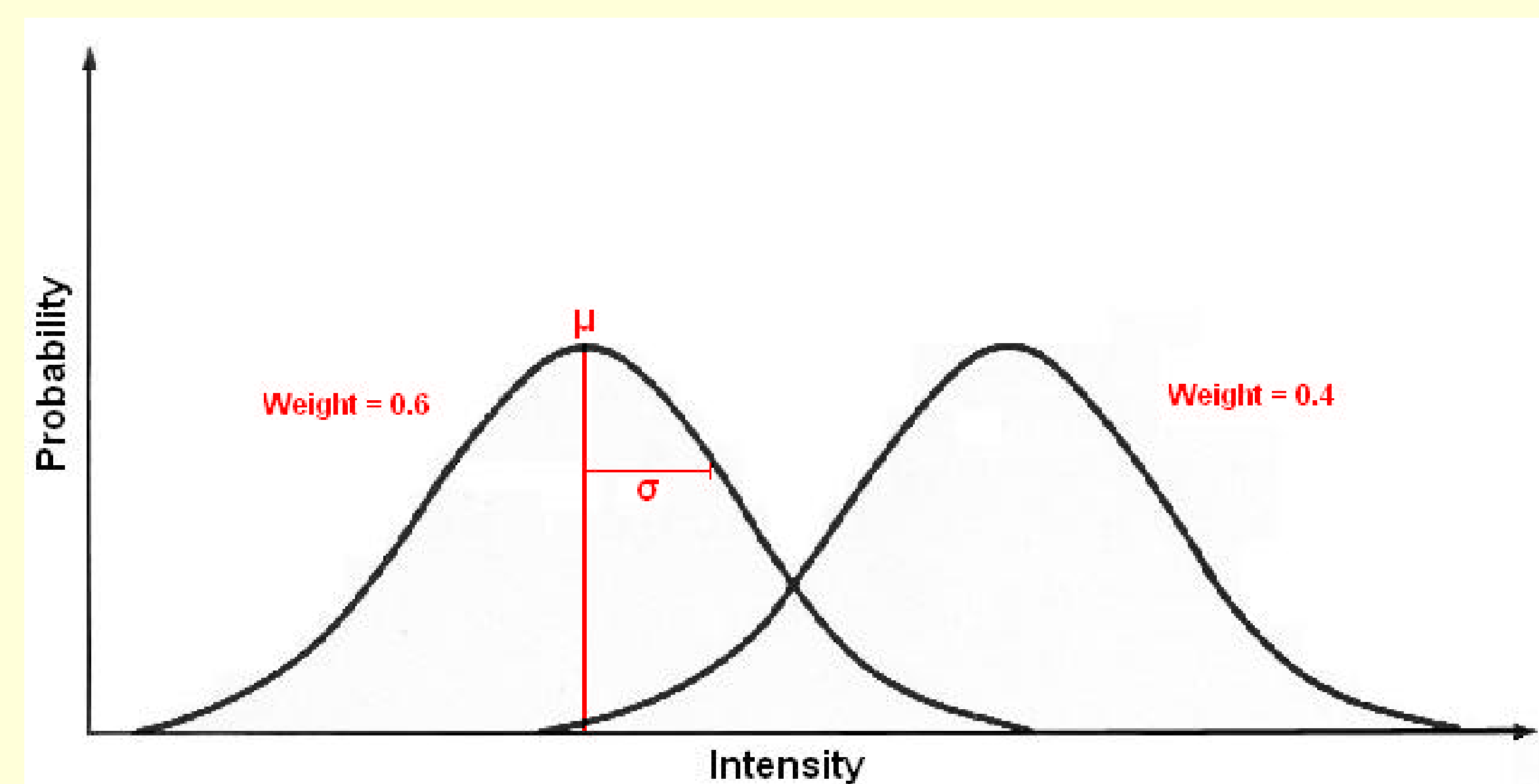

Diagram of how the Gaussian functions work

Filtering

Once the background is subtracted the noise needs to be filtered using simple morphological operators erode then dilate.

Blob Detection

With the filtered image the motion blobs could now be detected using the blob detection algorithm. This algorithm looks at each pixel and when it finds a blob it runs an inner boundary trace algorithm that finds the border of the blob, saves its data, and puts a blue box around it.
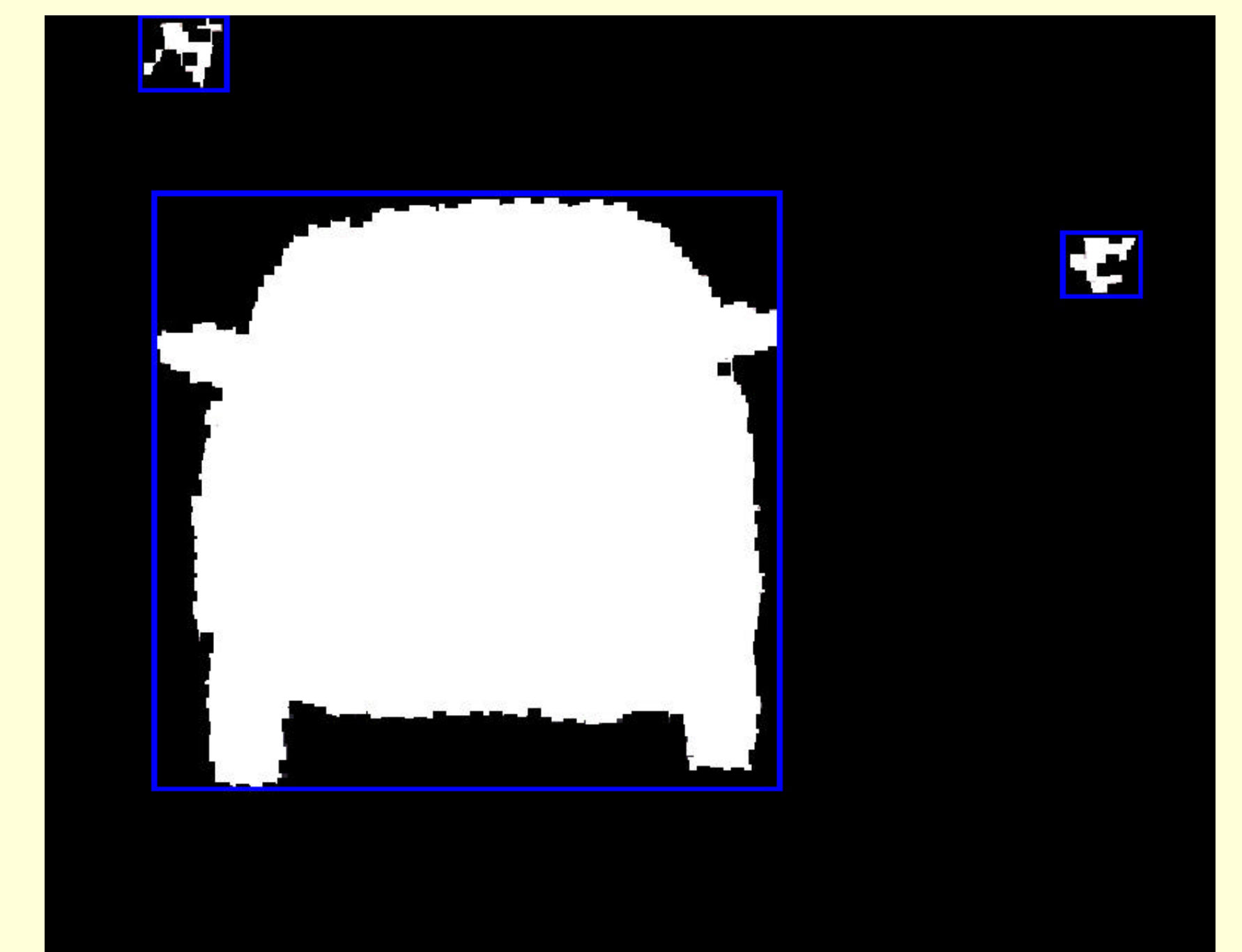
Visual output of blob detection

Chris Lankford

8-20-2004

Advisor: Hai Tao

SURF-IT

University California Santa Cruz

Jack Baskin School of Engineering