Tutorial on Range and Bearing Estimation

Basia Korel (bkorel@cs.brown.edu)

October 30, 2009

1 Introduction

Estimating the range and bearing from the robot to a landmark object is a necessary component for building a localization system to determine the robot's current pose. Specifically the range and bearing information is used to estimate the likelihood of a hypothetical pose of the robot given a map of the environment. This document serves as a reference to the *Tutorial on a Probabilisitic Measurement Model based on Landmark Range and Bearing Information*[1], which describes how to use the range and bearing information to estimate the likelihood function in the landmark measurement model.

The Probabilistic Measurement Model tutorial specifies a spotted landmark in the camera's visual stream as a *feature* and denotes it by the triple (r, ϕ, o) where r is the distance (range) from the robot to the fiducial, $\phi \; \epsilon \; (-\pi,\pi]$ is the angle (bearing) at which the fiducial was spotted and $o \; \epsilon$ {*yellow ball, pink fiducial, green/orange fiducial,...*} is its type. The bearing is 0 if the fiducial is directly in the middle of the robot's field of view. The bearing is positive if the landmark is to the left and negative if the landmark is the right of the robot's view center. This document describes how to determine the r and ϕ for each feature (r, ϕ, o). The estimated r and ϕ in the vector of observed features (1), where n is the number of objects in view, is then measured against the feature vector of a hypothetical pose to compute the likelihood that a hypothetical robot pose is the true robot pose.

$$f(z) = \{f_1, f_2, \ldots, f_n\} = \left\{ \begin{bmatrix} r_1 \\ \phi_1 \\ t_1 \end{bmatrix}, \begin{bmatrix} r_2 \\ \phi_2 \\ t_2 \end{bmatrix}, \ldots, \begin{bmatrix} r_n \\ \phi_n \\ t_n \end{bmatrix} \right\} \tag{1}$$

2 Measuring Landmark Objects

It is recommended that a data-driven procedure is used to learn a function that outputs the predicted range of landmark objects, each which have known dimensions and appearance, from perceived blob features. Thus the estimated distance to a landmark will be based on the dimensions of the associated blob(s) as seen through the robot's blobfinder. It is necessary to record the blob measurements for each landmark (pink fiducial, green/orange fiducial, orange/green fiducial, green goal and orange goal) in order to localize within the environment. If you choose to perform localization on the ball's location, then it is also necessary to measure blob dimensions for the yellow ball.

To estimate the range from your robot to a landmark, place each landmark at varying distances from the robot and record features (e.g., height, width, area) of the perceived blob(s) corresponding to the object, multiple times at each distance. You should record distances starting at 0 cm, or the closest the robot can see an object, to at least the entire length of the FC148 field. The result is a set of example input-output pairs that relate blob features to distance, as seen in Table 1. Note that measurements at far distances will vary only slightly and may become difficult to distinguish an accurate distance based on blob size.

[1] /course/cs148/pub/measurement_model_tutorial.pdf

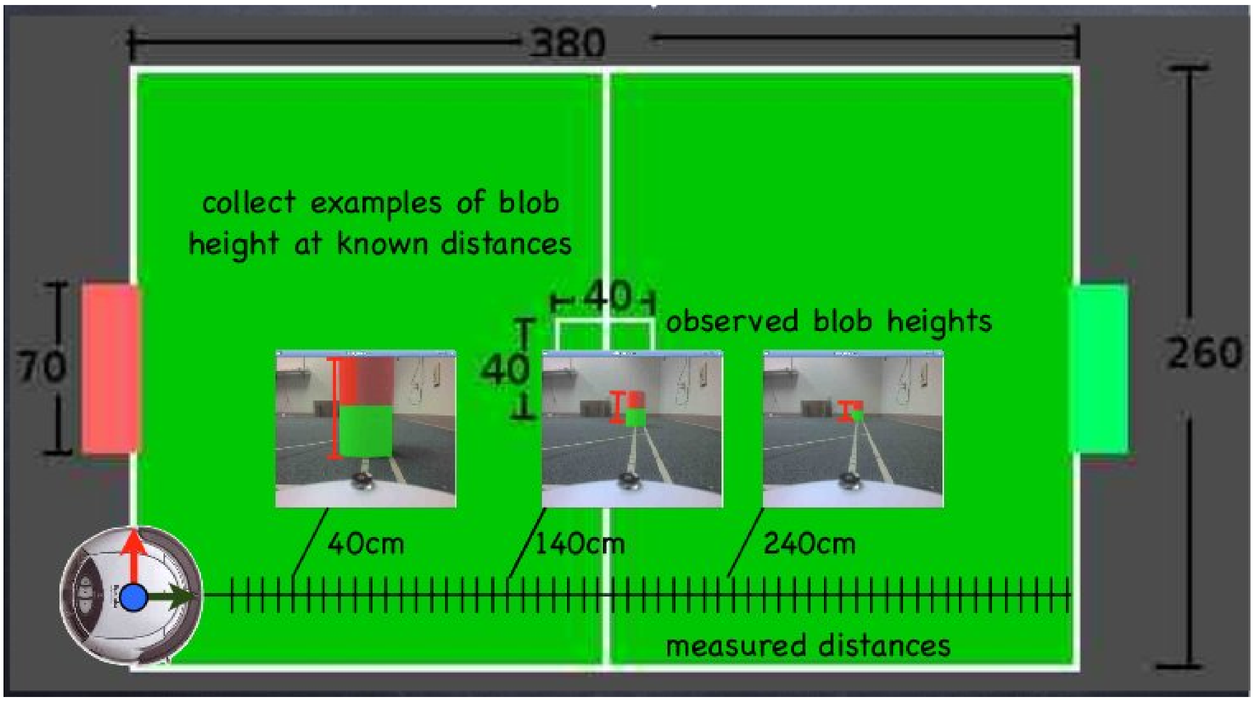

Figure 1: The process of collecting different blob dimensions for estimating range.

Table 1: Distance to height, width and area blob dimensions for the orange/green fiducial.

distance	height	width	area
25	239	144	29807
30	211	122	21241
35	206	114	20141
40	188	95	11527
45	167	88	9963

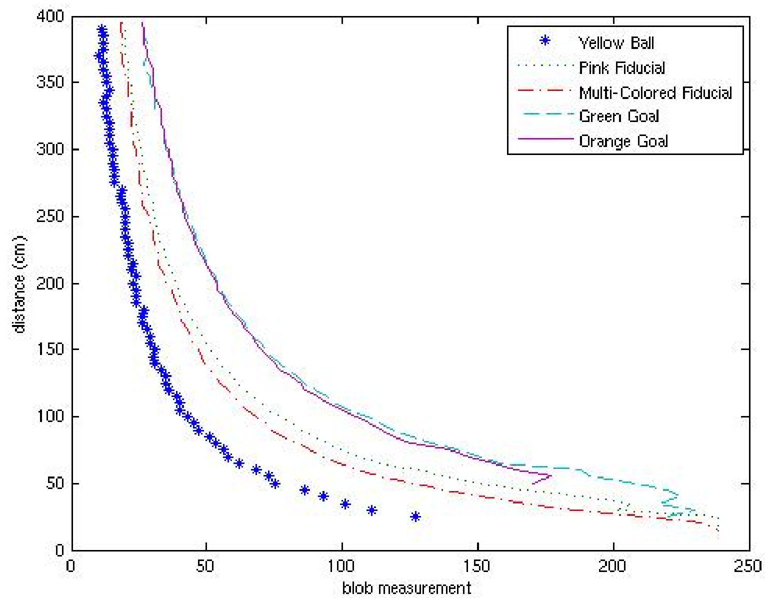

Figure 2: A plot of the distance over blob measurement for each object.

Figure 2 plots the distances over a selected blob measurement for each object. The blob measurement in the plot is the blob dimension the TAs felt most appropriate to use for each object. It is up to you to determine the appropriate features of blobs to use for estimating range. For different objects, certain blob features may be more reliable to use than others and it is helpful to plot the different blob features to determine this. For instance Figure 3 plots the blob height and blob width for one specific landmark. It is evident from this plot that the width is unreliable between approximately 60-105 cm. Such artifacts may occur because of the color calibration, lighting, or shape of the object.

3 Estimating Range

Once you have recorded blob measurements for each landmark, you should approximate the function that predicts distance from blob features. This blob-feature function can be approximated through a variety of regression techniques[2], including a nearest neighbors lookup table, linear interpolation, radial-basis interpolation, spline interpolation, and nonparametric regression. You should import your approximated function into your client. For example, a nearest neighbor regressor would import a lookup table data structure into a client. Upon seeing a blob, your client will first identify the type of object (ball, goal, landmark) for the blob, and then use the dimensions of the blob(s) to predict/lookup the distance to the landmark.

Figure 4, 5 and 6[3] are plots of predicting range from sample blob-distance data. Figure 4 is a nearest neighbor regressor (effectively a lookup table), Figure 5 is a RBF interpolator with scaled weights, and Figure 6 is a spline interpolator.

[2]Regression is the analysis of the relationship between a dependent variable(s) and an independent variable(s)

[3]The matlab example is located in /course/cs148/pub

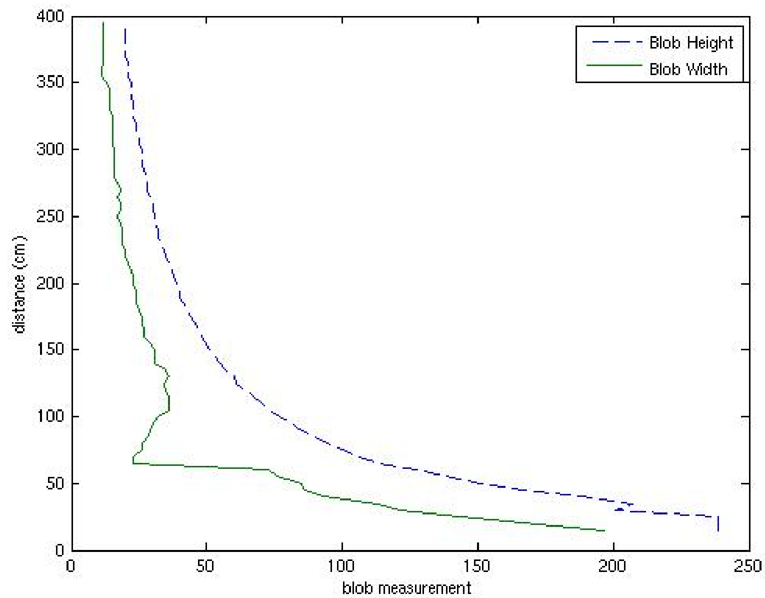

Figure 3: A plot comparing two different blob dimensions for estimating range.

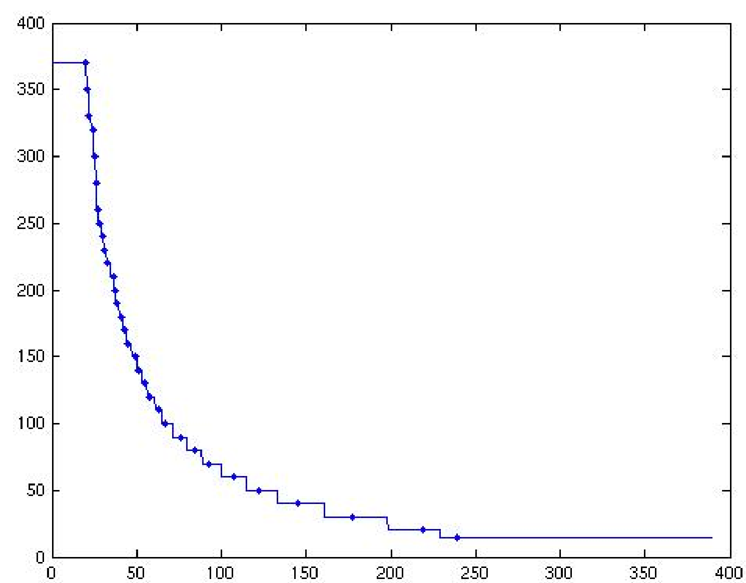

Figure 4: Range prediction using a nearest neighbor regressor.

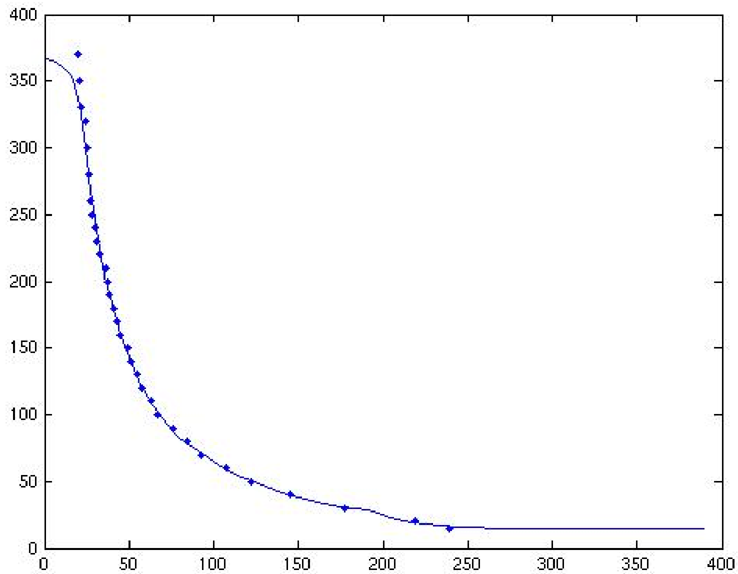

Figure 5: Range prediction using RBF interpolator.

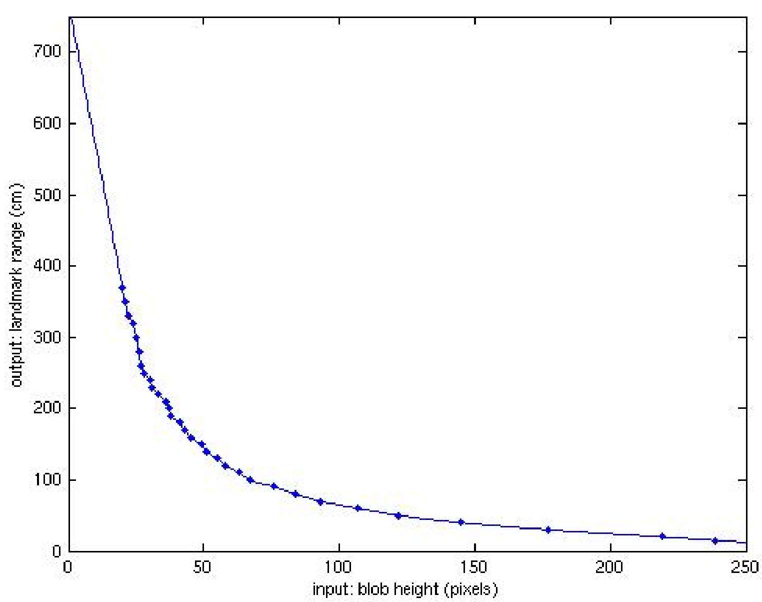

Figure 6: Range prediction using spline interpolator.

4 Estimating Bearing

To estimate bearing, it is suggested to use the relative proportions of the robot camera field of view, assuming the (default) image *view center* V_x is 160 pixels and half the camera *view angle* V_ϕ is 30 degrees. Refering to Figure 7, you can compute the bearing I_ϕ (in degrees) as follows:

$$I_x = V_x - blob\_center$$

$$I_\phi = \frac{I_x}{V_x} * V_\phi \tag{2}$$

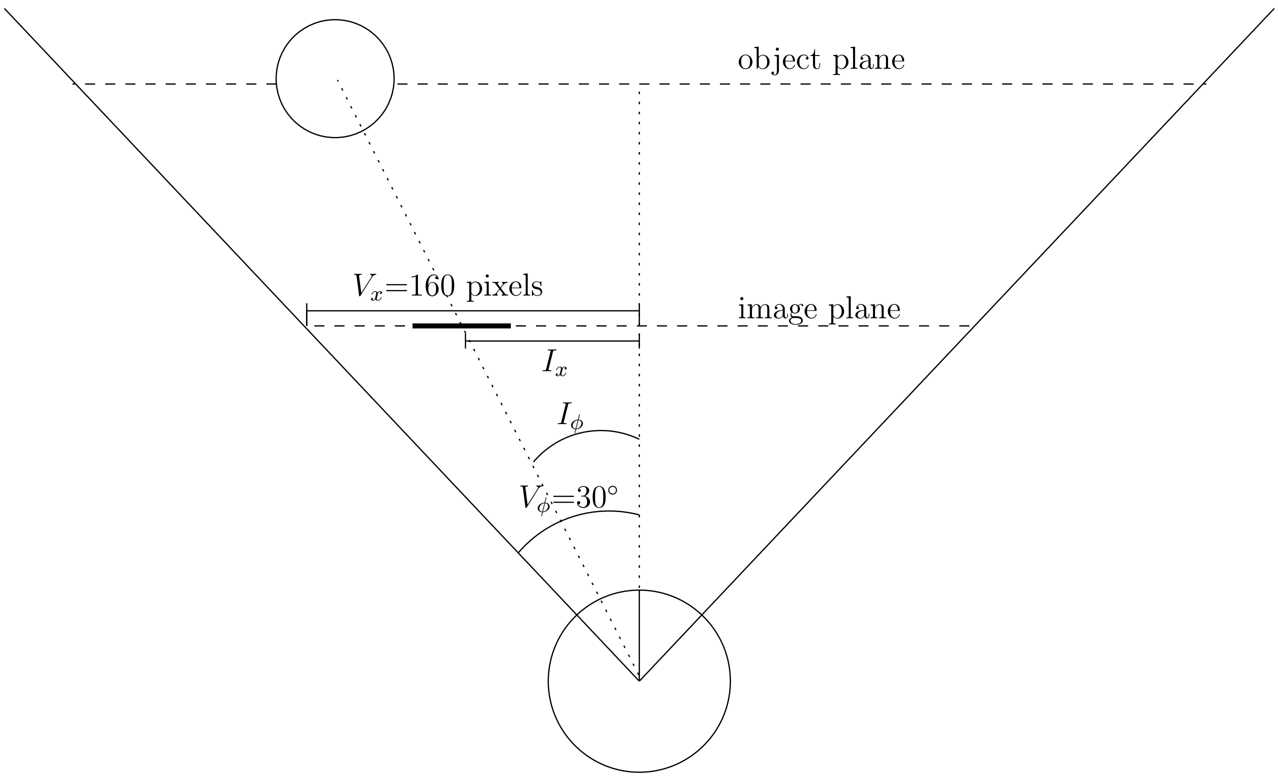

Figure 7: A landmark in the object plane projected onto the image plane as a blob. The estimated bearing I_ϕ is between a robot and the center of an object, assuming a 60 degree camera view angle and 320 pixel image width.