An Architecture for Online Semantic Labeling on UGVs

Arne Suppé, Luis Navarro-Serment, Daniel Munoz, Drew Bagnell and Martial Hebert

The Robotics Institute
Carnegie Mellon University
5000 Forbes Ave
Pittsburgh, PA 15213

ABSTRACT

We describe an architecture to provide online semantic labeling capabilities to field robots operating in urban environments. At the core of our system is the stacked hierarchical classifier developed by Munoz et al.,[1] which classifies regions in monocular color images using models derived from hand labeled training data. The classifier is trained to identify buildings, several kinds of hard surfaces, grass, trees, and sky. When taking this algorithm into the real world, practical concerns with difficult and varying lighting conditions require careful control of the imaging process. First, camera exposure is controlled by software, examining all of the image's pixels, to compensate for the poorly performing, simplistic algorithm used on the camera. Second, by merging multiple images taken with different exposure times, we are able to synthesize images with higher dynamic range than the ones produced by the sensor itself. The sensor's limited dynamic range makes it difficult to, at the same time, properly expose areas in shadow along with high albedo surfaces that are directly illuminated by the sun. Texture is a key feature used by the classifier, and under/over exposed regions lacking texture are a leading cause of misclassifications. The results of the classifier are shared with higher-lev elements operating in the UGV in order to perform tasks such as building identification from a distance and finding traversable surfaces.

Keywords: Semantic labeling, scene understanding, unmanned vehicles, computer vision

1. INTRODUCTION

Semantic labeling segments an image and labels regions so that they have meanings that are useful to higher level planning and scene understanding. This process provides important information upon which to make both tactical and strategic decisions. For example, in a field robot, this might help in discriminating among different kinds of traversable surfaces, each with physical properties that dictate a cost upon which a path planner optimizes a trajectory. In a military scout robot, semantic labeling can also identify buildings that affect the way a robot performs its task so as not to be detected.[2]

To perform this task, we use a method called Stacked Hierarchical Labeling by Munoz et al.[1] This method first segments that image into a hierarchy of regions. The regions are classified from coarse to fine, with the coarse levels' results passed to their children as evidence of the label distribution expected in that local context. In this way, the model captures spatial and relational information about a scene. A sample result is in Figure 1. In this case, the classifier performs well, mislabeling a few façade pixels as object and some tree in the distance as building. Like all scene labeling systems, basic features provide evidence about the class of an object. In this system's case, these feature are SIFT keypoints and various texture related measures. While not directly affected by image brightness, all are based on some derivative of the pixel values. SIFT is dependent on the Laplacian of the pixels and the texture measures are essentially linear filter responses.

When images are under or over-exposed, the measures are not distorted in some regions of the image because of clipping effects or completely degenerate. Textureless white regions in an outdoor environment are usually sky regions, but may also be overexposed regions, and when integrated into the hierarchical inference construction, can cause non-local labeling errors. In figure 2, the overexposed sun-facing façade was incorrectly labeled as sky.

Further author information: (Send correspondence to A.S.)
A.S.: E-mail: suppe@ri.cmu.edu, Telephone: 1 (412) 268-6034
L.N.: E-mail: lenscmu@ri.cmu.edu, Telephone: 1 (412) 268-6034

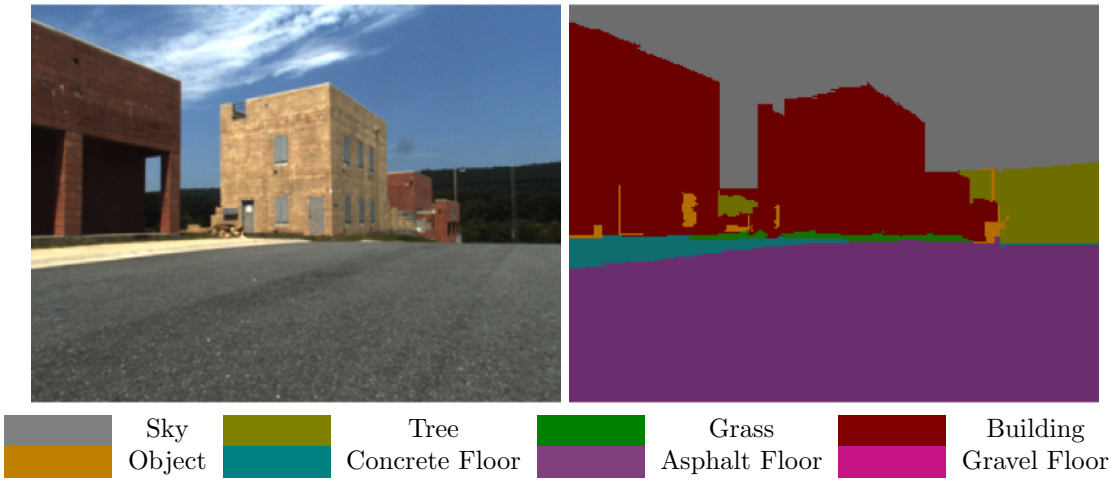

Figure 1. Left, source image. Right, output of semantic classifier

While the algorithm performs rather well considering the poor image quality, the sky to the right side of the label image is strangely also labeled as building. A natural way to solve this problem is to combine overexposed and underexposed images and to somehow produce a composite image out of regions selected from the source image with best exposure. While this might distort the image content somewhat, we note that the features upon which the algorithm is based are not dependent on the absolute intensity of the pixels.

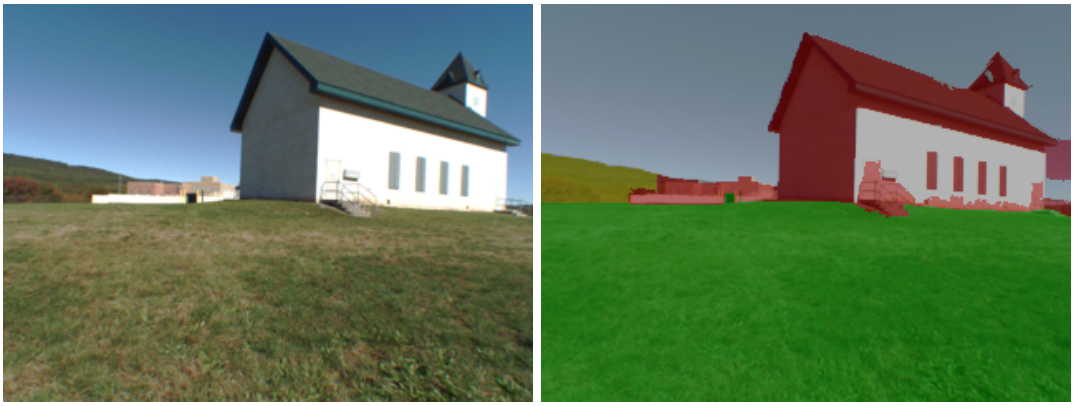

Figure 2. Left, source image with overexposed regions. Right, output of semantic classifier

2. IMAGE ACQUISITION

This section describes the process used to acquire and prepare the images that are fed to the classifier. The objective is to capture images that are as informative as possible, i.e., so that textures can be perceived. Images which are either under- or over-exposed usually contain textureless areas in which details are lost due to the incorrect exposure. Areas in the image that are either too dark or too bright do not contain enough information, which reduces the labeling accuracy. Conversely, high dynamic range (HDR) images (i.e. images in which the luminance range between the lightest and darkest areas of the image is larger than in a conventional image) usually avoid these extremes,[3] and therefore contain more information, which improves performance. For example, consider a set of pictures taken inside an office building, as shown in Figure 3. A short time of exposure (left) is adequate for imaging the building seen outside the window. However, the area corresponding to the inside of the office is so dark that is barely perceived. A long time of exposure (center) has the opposite effect: the inside of

the office is captured correctly, while the building is lost. Finally, elements both inside and outside of the office can be perceived more easily in a HDR image of the same scene (right).

Figure 3. Left, short exposure. Center, long exposure. Right: high-dynamic range.

HDR images are generated by combining images of the same scene captured with different times of exposure, in a process known as *exposure bracketing*. The HDR image shown in Figure 3-right was produced by combining the short and long exposure images in the same figure. For our applications, although HDR cameras are commercially available, we decided to implement our system using a regular CCD camera, mainly to have more control of the generation of HDR imagery. Furthermore, the particular constraints of the experimental platform used in this study made it difficult to find a suitable off-the-shelf HDR camera.

For our purposes, we needed a technique capable of increasing the dynamic range, at a low computational cost, and using a minimum of input images. Several approaches to combine multiple images into a single HDR picture have been described in the literature.[4,5] Most of them include a tone mapping step, which is used to approximate the appearance of the HDR images when displayed in a medium that has a more limited dynamic range.[6] This process consumes valuable computing resources. Furthermore, it is not necessary in our application, since we are only concerned about the robust and consistent extraction of SIFT keypoints and other texture related measures. Therefore, the tone mapping step is not carried out in our system. Similarly, we conducted a series of tests to determine how many images were needed to generate a suitable HDR image. It was found that only two images were enough to increase the performance; a larger number of images did not improve the results significantly. Consequently, to keep computational costs low, we decided to use only two input images.

Our implementation is based mainly on the work by Gelfand et al.,[4] and was created around a Basler™ACE1300-30gc camera with a 1296×966 1/3" CCD sensor, with a GigE interface.

The generation of HDR images involves a sequence of steps, which include: 1) Determine base exposure value, 2) Calculate exposure times for high dynamic range (i.e. limits for exposure bracketing), and 3) Merge into a single high dynamic range image.

2.1 Base exposure value

In this step, the average luminance of the current scene is calculated. This provides a reference for the capture of two subsequent images with different times of exposure. For a set of camera settings $\{F, T\}$, the corresponding exposure value EV is given by

$$EV = \log_2 \frac{F^2}{T} \tag{1}$$

where F is the aperture size and T is the duration of the exposure. This combination of aperture and shutter speed produces an image with an average brightness B_{pre}. Assuming that the apperture F remains constant, we focus on adjusting the time of exposure to increase or decrease the brightness of the images captured. To this end, we calculate EV_{opt}, which is the exposure value that would result in an image with a desired average brightness B_{opt}. This is computed using the expression

$$EV_{opt} = EV_{pre} - \log_2 (B_{opt}) + \log_2 (B_{pre}) \tag{2}$$

The value chosen for B_{opt} is typically obtained by comparing against an image of a 18% gray calibration card. By substituting EV_{opt} in equation(1), the corresponding time of exposure T_{opt} is obtained:

$$T_{opt} = 2^{(\log_2 F - EV_{opt})} \tag{3}$$

In our application, this time of exposure is used as a reference, and to validate whether current conditions are favorable for collecting images. For instance, times of exposure that are outside a certain range of values may indicate extreme conditions that will result in poor performance (e.g. too dark, or camera is facing directly to the sun).

2.2 Exposure bracketing

Once that EV_{opt} has been determined, we proceed to calculate the times of exposure that will produce the two input images. Let us define $EV_{long} = EV_{opt} + \delta_{long}$ and $EV_{short} = EV_{opt} + \delta_{short}$, which indicate the exposure values for long and short times of exposure, respectively. These values are obtained by shifting EV_{opt} by δ_{long} and δ_{short} stops respectively, where $\delta_{long} \geq \delta_{short}$. The corresponding times of exposure are calculated in the same way as T_{opt} :

$$T_{long} = 2^{(\log_2 F - EV_{long})} \tag{4}$$

$$T_{short} = 2^{(\log_2 F - EV_{short})} \tag{5}$$

In our system, a series of tests showed that images collected at $\delta_{long} = -1$ and $\delta_{short} = -3$ stops from EV_{opt} produced the best results, in terms of merging images. These tests consisted of collecting sets of images where the exposure values were bracketed from -3 to +3 stops from EV_{opt}, in increments of one stop, at different times of the day (e.g. morning, noon, and afternoon), and under different weather conditions (e.g clear sky, partly cloudy, overcast). Then, the image entropy[7] was calculated for images merged using different combinations of pairs of exposure values. We used entropy as a measure of the contrast in the image, where a higher entropy value denotes a higher contrast. On average, the images merged by combining pairs with exposure values $\{-3, -1\}$ were found to have the higher entropies.
The two images are captured within a few milliseconds from each other. It is important to note that the camera should not move, to facilitate the simple merging algorithm described in the following section. This was not a problem in our application, since the robot was commanded to stop for image acquisition. However, there are approaches[5] to generate HDR images that can be used while the camera is in motion*.

2.3 Image merging

The long and short exposure images are combined into a single HDR image by computing a scalar-valued weight map for each image, and then performing a weighted mixture. Given a pair of input images P_{long} and P_{short} captured with the exposure times T_{long} and T_{short}, respectively, where the luminosity of each pixel is stored in the arrays $I_{long}(i,j)$ and $I_{short}(i,j)$. The weight of each pixel according to its luminosity is calculated as:

$$W_k(i,j) = \exp\left(-\frac{(I_k(i,j) - \mu \cdot 255)^2}{2(\sigma \cdot 255)^2}\right) \tag{6}$$

These weights are calculated for each image, resulting in the arrays $W_{long}(i,j)$ and $W_{short}(i,j)$. These arrays are normalized, so that the sum of values from both images for every pixel equals 1. In our system, the parameters were set as $\mu = 0.5$ and $\sigma = 0.2$.

Finally, the input images are merged using the expression

$$R^q(i,j) = W_{long}(i,j) \cdot P^q_{long}(i,j) + W_{short}(i,j) \cdot P^q_{short}(i,j) \tag{7}$$

where q represents each channel (i.e. color component) of the input images. The final HDR image, P_{HDR}, is the union of all the R^q channels.
P_{HDR} is rectified and converted to a 320×240 size before entering the semantic labeling module.

*The ability to generate HDR images as the robot moves will be implemented in future revisions.

3. EXPERIMENT

We trained our classifier on 438 images of which only 138 used the HDR technique. The remaining used a fixed exposure and aperture. Ideally, this experiment would train using images captured under identical conditions to those presented to the classifier. However, since the cost to hand label data is high and our immediate goal was to improve classifier performance using all the training data, we instead show here that the HDR technique performs superior to any single exposure image when measured against a labeled testing set of 265 images (Table 1). These images were captured in quick succession (about 15 Hz) under a variety of exposure levels, so the images are essentially identical. The macro-averaged F1 score is used to compare the performance obtained with different exposure settings.[?]

Table 1. $F1_{macro}$ for classifier when tested against images taken at various exposure settings. The HDR images, a combination of F/+1 and F/-1, was significantly superior in performance to any one exposure setting.

Effective Exposure Setting	$F1_{macro}$
F/-3	0.622
F/-2	0.723
F/-1	0.720
F/+0	0.632
F/+1	0.404
HDR	**0.850**

4. CONCLUSIONS AND FUTURE WORK

We presented an implementation of a classic technique for increasing the dynamic range of a camera by combining images taken at different exposure settings to improve the performance of a state-of-the-art semantic labeling classifer. In this way, we are able to ensure that few regions of an image are either over-exposed or under-exposed, and that the image has texture in shadow and bright sunlight. While this technique is computationally very simple, it is not without drawbacks.

Even though we can manipulate the camera's exposure settings at 15 Hz, robot motion will cause a significant mis-registration between the stacked images. While the robot's linear motion in that period is small, the camera's motion when traversing bumpy terrain is significant. Optical flow techniques can estimate the camera motion and realign the images, provided that 3D parallax effects are small. For a scene classifier such as ours, this is often the case.

Unlike more advanced techniques, all the regions in each input image are combined linearly. This means that an image with long exposure, intended to capture detail in shadow, may have saturated pixels in brighter regions. When combined with a short exposure image, these saturated pixels may still wash out regions in the image. For this reason, there is still a limited range of brightness that can be captured with this technique. The classic solution to this problem is to combine a mechanical iris and electronic exposure control systems which dynamically adapt to the environment. While our technique manipulates electronic exposure control with a fixed aperture, neither can overcome the fundamental physical limitations of the sensor mechanism itself.

We will soon integrate our classifier with a true high dynamic range digital camera system, custom built by the National Robotics Engineering Center at Carnegie Mellon University. The sensor is a Pixim SeaWolf imager and features a per-pixel exposure control system rather than a global shutter. These sensors are typically integrated into low-cost security cameras designed to operate in bright sun or in darkness without mechanical irises. As such, the available resolution is limited to just 640×480. These cameras have only been available commercially with analog output. Our custom all digital system is designed for tight integration with vehicle state systems and a 3D mapping LIDAR for point cloud colorization. Figure 4 demonstrates how the camera captures detail in shadow and light when our standard machine vision camera can not.

Figure 4. Left, sample image from Pixim SeaWolf captures detail in all regions, as compared to the Basler ACE camera, right. In particular, compare the floor where the shadows from the garage door are cast.

5. ACKNOWLEDGMENT

This work was conducted through collaborative participation in the Robotics Consortium sponsored by the U.S Army Research Laboratory under the Collaborative Technology Alliance Program, Cooperative Agreement W911NF-10-2-0016.

REFERENCES

[1] Munoz, D., Bagnell, J. A., and Hebert, M., "Stacked hierarchical labeling," in [*Proc. ECCV*], (2010).

[2] Oh, J., Suppe, A., Stentz, A., and Hebert, M., "Enhancing robot perception using the eyes of human teammates," *Autonomous Agents and Multiagent Systems AAMAS* (2013).

[3] Robertson, M. A., Borman, S., and Stevenson, R. L., "Dynamic range improvement through multiple exposures," in [*In Proc. of the Int. Conf. on Image Processing (ICIP99*], 159–163, IEEE (1999).

[4] Gelfand, N., Adams, A., Park, S. H., and Pulli, K., "Multi-exposure imaging on mobile devices," in [*Proceedings of the international conference on Multimedia*], *MM '10*, 823–826, ACM, New York, NY, USA (2010).

[5] Kang, S. B., Uyttendaele, M., Winder, S., and Szeliski, R., "High dynamic range video," *ACM Trans. Graph.* **22**, 319–325 (July 2003).

[6] Qiu, G., Guan, J., Duan, J., and Chen, M., "Tone mapping for HDR image using optimization a new closed form solution," in [*Pattern Recognition, 2006. ICPR 2006. 18th International Conference on*], **1**, 996–999 (2006).

[7] Sonka, M., Hlavac, V., and Boyle, R., [*Image Processing, Analysis, and Machine Vision*], Thomson-Engineering (2007).