Webcam Configurations for Ground Texture Visual Servo

Takeshi Matsumoto, David Powers, Nasser Asgari
School of Informatics and Engineering
Flinders University
Adelaide, Australia
takeshi.matsumoto@flinders.edu.au

***Abstract*— Increase in accessibility of low cost webcams has lead the way for hobbyists and researchers to include affordable, yet information rich, visual sensors for areas such as machine vision. This work investigates the practical considerations of using a floor pointing visual sensor for visual servo applications. Several conditions and characteristics of both the webcam and the ground texture tracking are explored to provide efficient use of the camera. Using the discovered characteristics, several image restoration filter algorithms are found to improve the signal to noise ratio with good efficiency and efficacy in ground texture tracking applications. The findings here form the basis for a localization algorithm for a mobile robot platform.**

***Keywords*—Visual servo, image processing, localization**

I. INTRODUCTION

Visual data processing techniques has steadily increased in popularity due to the continued improvements in the quality and the accessibility of visual sensors. Off-the-shelf webcams have become popular additions for hobbyists and researchers interested in integrating visual data to various projects, such as surveillance and machine vision. One particular application within the field of mobile robots [1] is in vision based localization algorithms [2] [3]. Using the vast repertoire of coordinate transformation and feature tracking algorithms, the task of translating visual information to positional knowledge is becoming better understood.

In a typical configuration, the camera sensors can observe the environment from many different points of view to make use of the rich information that is available, but by compressing the 3D environment into a 2D view, ambiguity is introduced which increases the complexity and the misalignment between the environment and the internal representation. For this reason, camera sensors are sometimes known as directional sensors, as the orientation of obstacles are projected onto the image plane. Acknowledging this characteristics and using the camera to only expect 2D information, can lead to simplification of visual processing algorithms, as well as more accurate translation of the visual data. One widely used application of this technique is in the technology and approach used in an optical mouse, which tracks ground texture displacements from a predetermined perspective. The proposed visual servo approach builds upon the same idea, but instead of using the purpose-built optical sensor used on the optical mouse, it uses an off-the-shelf-webcam [4].

A critical assumption made in ground texture tracking is that the distance from the camera to the ground is fixed and known. This assumption then allows a simple triangulation process to measure the motion and provides accumulative relative localization information. Purpose built hardware allows very precise motion detection, but at the cost of restrictions placed on where it can be used. The major difference in using the webcam lies in the adaptability for different environments at the cost of precision, along with software processing involved in the computation. The image acquisition speed on an optical mouse is typically a few order of magnitude faster than a typical webcam and provides a very precise view of the ground texture, but is very limited in the type of environments that it can be used on, as well as the inability to distinguish significant features due to the small viewing area.

This study focuses on the practical issues surrounding the use of a webcam for mobile robot localization and investigates filtering algorithms to improve the quality of the streaming images.

II. CAMERA SETTINGS

The placement of the cameras plays a very important role in determining the precision and accuracies of the feature tracking algorithm. It is also indirectly responsible for issues such as lighting, frame rate, focal distance, tolerance in height changes, and operating speed of the robot. In this section, many practical issues are discussed with regard to the characteristics of the camera and the anticipated captured data.

A. Mount position

Instead of applying coordinate transformations after calibrating the characteristics of the camera, a well controlled placement of the camera allows reduction of unnecessary transformation of the data. In the absence of adaptive camera configuration during runtime, this approach allows better utilization of the calibration process. With this in mind, the camera was aligned with the robot's coordinate axes and positioned parallel to the ground so that there was a constant mapping between the axial changes.

The viewing angle, which is one of the most important parameters used in translating the observed motion to camera motion, can be calculated from two or more measurements made at known heights. Due to the slight differences in the vertical and horizontal viewing angles, two coefficients were stored to treat each axial displacement separately.

The typical viewing angle of a webcam is quite small at around 40 degrees horizontally and 30 degrees vertically, which creates a very small viewing area especially when the camera is placed close to the ground to distinguish the ground texture patterns. This limitation places a restriction on the maximum velocity of the robot, since the ground textures need to appear in subsequent frames for the displacement to be detectable. The height of the camera can modify the viewing area, allowing control over the amount of displacement along with the operational speed of the robot. Although the level of detail which can be detected will decrease, the increase in the camera height allows a larger, more distinctive pattern to be found. A portion of the view by a typical ground texture observed from a camera mounted 80 millimeters off the ground can be seen in Fig. 1.

A related issue to consider is the effect of increased operational speed of the camera or its resolution setting on the processing load. By increasing the speed, the feature search area increases while an increase in the resolution can often overload the processor, which can result in the loss of frames or compensation by reducing the viewing area. Rather than simply restricting the operation speed or availability of data, a careful balance between the speed, precision, and the processing load is required to suit the environment and task the robot is performing.

The critical assumption that the camera position relative to robot and the distance to the ground stays consistent throughout the operation means that any changes in the height, such as those caused by bumps or slopes, causes drifts to occur in the measurement. Low cost solutions such as soft tyres to absorb small bumps or the use of specialised sensors such as gyroscopes and accelerometers can be implemented to reduce some of these errors. If focus control is available, this feature can also be used to detect and reduce the displacement errors at no extra hardware cost.

Figure 1. Typical ground texture. Carpet on the left and table on the right

B. Lighting

The assumption that the distance between the camera and the object should remain constant requires an additional assumption that obstruction of the ground should never occur. However, this assumption does not stop shadows from occurring which can cause significant changes in the ground texture appearance. The passive nature of the device also means that a light source is essential for its operation. To compensate for these issues, an additional light source can be used in conjunction with the camera to provide a consistent non-obscured light source. This also prevents potential biasing or noise from changes in the ambient light conditions and increases the consistency of ground texture appearance.

The brightness, shape and the direction of the light has to be carefully selected such that the ground textures are not saturated, nor biased from a particular orientation. Directional lighting allows greater exposure of the rough surface texture, but the biased enhancement limits its usefulness to the same viewing angle. To allow generic features to appear and to reduce the orientation dependency, lighting should be provided as evenly as possible over the image from the top. To achieve this uniformity, several light sources can be placed around the camera to provide even lighting, or scattered light from a Lambertian surface can be used if a bright light source is available. For similar reasons, a white light source should be chosen to maintain the rich variety of intensities of the ground texture. This allows a higher level image processing task to control the amount of information it requires.

The use of LEDs as the light source resulted in distinctive spot patterns appearing due to the limited number of LEDs and the localized brightness of the light. Hence, an alternative approach was used in which a small torch was mounted above the camera at an angle and directed to a crumpled sheet of aluminium foil to provide even refected light.

C. Exposure time

Most webcams are equipped with automatic control to adjust the exposure time and gain to suit the ambient light present in the environment. This is often a very useful feature for smoothing and still images, but it can also cause undesirable intensity shifts and motion blur.

Ideally, the exposure time should be set very short to increase the tolerable displacement of objects while the intensity is being captured. However, this must be balanced with the available light since reducing the exposure time also decreases the colour richness of the captured texture. With the introduction of the permanent light source, the exposure time can be controlled more readily by changing the brightness of the light. To decrease the exposure time, the light needs to be quite bright and uniformly distributed, as the variation in brightness is greater. Manually decreasing it slightly to reduce motion blur yields more benefits than the small loss in the intensity range due to the importance of edge clarity.

A problem occurs for fixing the exposure time when the ambient light conditions change dramatically. This is typically caused by shadow for outdoor and the flickering effect caused by the room lights with the AC induced timing differences for indoor operations. The overall change in the intensity caused by the flickering can be quite significant, but most of the relative texture information remains consistent that allows short term features to be identified and used. However, this requires a more complex processing, such as waiting for the brightest moment, when capturing a long term feature. A simple solution to reduce the effect of ambient lights is to shield the view area and only allow the consistent light source to be used.

D. Sample rate

The final camera setting to be considered is the capture rate of the camera. Similarly to camera height, this plays an equally important role in specifying the accuracy, variance in speed, and the amount of data to be processed. One of the key contributors in the precision achieved by an optical mouse is its extremely fast frame rate, which can be over 60 times that of a typical webcam. Although the increased volume of data increases the processing load, the small time differences between the frames allows smoother and smaller transitions to occur, thus decreasing the search area. For this reason, it is useful to use the maximum frame rate when possible and offset the increase in the data through optimized or selective processing of relevant data at a higher level.

The timing loops controlling the updating of the memory which holds the image data and those controlling the reading of the data are often separated and require synchronization. If the read operations are slower than the frame rate of the webcam, the excess data will typically be discarded, which can cause incorrect accumulation, while if the localization loop reading the image is too fast, duplicate images could be read, which can cause discrepancies with the expectations. Timers or semaphores can be used for synchronization. However, a critical condition that must always be adhered to is that the feature to be tracked must be visible in the subsequent frame. Failure to meet this causes unrecoverable errors where an incorrect displacement is detected. Hence, the localization loop must always remain equal to or faster than the sample rate for the webcam where a potential work around is to buffer the image data until they can be processed. Once this is achieved, since the probability of encountering the exact same sequence of intensity is extremely low due to the hardware noise, only a simple comparison between the current and the last frame is required to detect and discard the duplicate data.

III. Image Characteristics

The areas of image processing [5] and robotics have often progressed in parallel to each other, especially due to the variety of applications that the rich information can be used in. The increase in accessibility of low cost visual sensors allows simple access to enormous amount of information about the environment which can potentially be converted to added knowledge for the robot. Even with constraints to narrow down the different interpretation of data, the internal representation of the environment does not always align with the real world due to misinterpretations or corrupted sensory data. A typical off-the-shelf webcam is equipped with a CMOS chip which is known to capture noisy data due to the configuration of the photo-sensors and photo transistors that are typically arranged in a fixed pattern to approximate a colour. The use of plastic lenses also contributes to a blurred image, which can increase the misalignment of the internal representation. Some of these noises can be reduced by calibration process and domain knowledge of the type of images that is expected to be seen.

In the following section, many characteristics of the captured image from the webcam are investigated and the conditions and effects they cause on the captured data are observed.

A. *Lens distortion*

Many off-the-shelf webcams are fitted with inexpensive lenses that do not provide uniform mapping between the actual view and the image on the projection plane. The lens distortion near the outer edges can be reduced by measuring the amount of distortion, typically by taking a snapshot of a known scaled grid, then fitting a transformation matrix to re-align the grid correctly.

The camera used in the current implementation showed very little warping due to the manufacturer's lens distortion correcting algorithm, which was verified using line finding algorithms, as per Fig. 2. However, the side effect of this approach caused obvious sign of blurring at the outer edges of the image. To avoid further processing which would have minimal improvements on correcting the distortion, the outer strips which showed visible sign of blurring were simply cropped out for the subsequent analysis. The reduction of the viewing area does not have a significant effect on the visual servo algorithm, as only a portion of the view will be used to reduce the computational load. A useful characteristic of the warping is that the amount of distortion is gradual, thus small displacements within the viewing area will result in very small distortion between the consecutive frames.

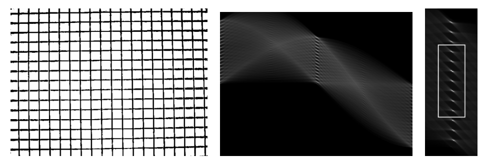

Figure 2. Lens distortion analysis using Hough transformation

B. *Random noise*

The importance of noise reduction in the proposed visual servo approach is related to the repetitive nature of the ground texture as well as the small number of variation in the intensity at a particular instant. As the majority of noise cannot be prevented at the source level, such as hardware induced noise, suppression filters must be employed to reduce and recover from the corrupted data.

To obtain the typical range of variation in the intensity due to random noise, the camera was exposed to a uniform, constant intensity to measure the minimum, maximum, mean and the standard deviation characteristics for each pixel. This initial analysis was carried out to determine the individual pixel's signal to noise ratio. The sampling colours were set in a grey scale range to avoid interference from neighbouring photo-sensors. Uniform intensity was the easiest to achieve with the two extreme colours, as all it required was full saturation and the absence of light. However, a uniform grey could not be achieved as accurately due to other noise factors and sensitivities. The inaccuracies resulted in higher variance, but the trends at various intensities could still be illustrated. The calibration measurements were terminated once the mean value converged after the number of samples was greater than the range of values it could distinguish.

Fig. 3 shows the standard deviation trends at different intensity reading for each of the three colours. Rather than storing the individual rate of fluctuation, the trend can be modeled by a polynomial function to save the memory footprint and lookup time at the cost of performing the conversion dynamically. With R, G, and B representing the intensity reading of each colour between 0 and 255, the conversion functions for the camera used in the current implementation were found as follows.

$$StdDev_R = -5\text{x}10^{-7}.R^3+3\text{x}10^{-4}.R^2-5.51\text{x}10^{-2}.R+6.0298 \quad (2)$$
$$StdDev_G = -1\text{x}10^{-6}.G^3+6\text{x}10^{-4}.G^2-9.06\text{x}10^{-2}.G+6.3492 \quad (3)$$
$$StdDev_B = -9\text{x}10^{-8}.B^3+2\text{x}10^{-4}.B^2-5.18\text{x}10^{-2}.B+7.1179 \quad (4)$$

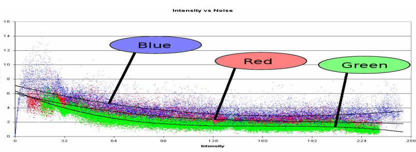

Figure 3. Intensity vs standard deviation trend

An interesting behaviour which was observed was that no noise was present once the saturation point is reached. Although cleaner data is preferred in terms of noise reduction, biased image where everything is made brighter results in many saturated sections and a reduced variation, thus misses out on the richness of the texture. It is worth noting that the saturation point is dependant on the camera settings such as brightness and gain, thus setting these appropriately will maximize the variation in intensity.

C. *Colour based noise*

To determine the influence of neighbouring colour sensors, the camera was exposed to a set of intensities of varying hue and saturation to record the change in intensity, as shown in Fig. 4. Not all of the available colours were investigated in detail because of the huge possible variations, but simple tests revealed a visible trend, thus prompting further analysis.

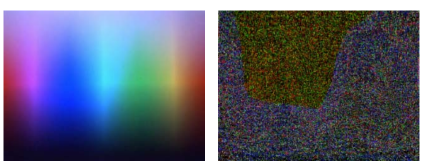

Figure 4. Colour based normalised standard deviaion

A more detailed analysis involved a similar approach, but a smaller sampling window was used to allow simultaneous analysis of several colours at once. The placement of the smaller window was manually done to avoid noise prone areas which were detected in the previous analysis. By reducing the precision in differentiating the intensity, it allowed enough reduction in the variation for a colour to noise lookup table which was used in conjunction with simple linear interpolation to account for the missing colour values. If I, I_0, I_{MAX}, and P represents the scaled intensity, the original intensity, the maximum value of intensity and the precision respectively, while M is the lookup table for the standard deviation value, the interpolated standard deviation can be derived from

$$I = I_0 / I_{MAX} \text{ x } P \quad (5)$$
$$StdDev = (I - \lfloor I \rfloor) \,.\, M_{\lfloor I \rfloor} + (1 - I + \lfloor I \rfloor) \,.\, M_{\lfloor I \rfloor + 1}. \quad (6)$$

Considerations such as the arrangement of the photo-sensors are not discussed here, but analyzing the particular arrangement can potentially give rise to better filtering which exploit the photo-sensor to colour relationships.

D. *Radial intensity shift*

An additional characteristic observed by the noise analysis for individual photo-sensors was the trend in the average intensity values around the image. It was observed that the average intensity of the inner positions tended to be higher than the outer positions, as shown in Fig. 5. This behaviour could be attributed to issues such as the lens characteristics, the inverse square law of light intensity distribution, or secondary rays from reflections. This characteristic can be suppressed to a degree from the cropping done to reduce the lens distortion, but the trend can be modeled to identify the intensity shifts at different positions in the image.

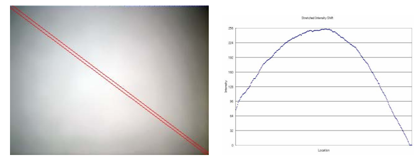

Figure 5. Stretched radial intensity shift with slice of the intensity.

After determining the center of the shift, a polynomial function can be derived to determine the amount of adjustment required to flatten the image. The function derived for the current camera showed the relationship below, where i and j represent the pixel coordinate point, i_c and j_c represents the center point of the radial shift, and D represents the distance from the center point.

$$D = \sqrt{((i - i_c)^2 + (j - j_c)^2)} \quad (7)$$
$$Shift = D.(1\text{x}10^{-3}.D + 6.3\text{x}10^{-3}) \quad (8)$$

IV. IMAGE FILTER

The observations from the various characteristics made during the calibration stage can form the basis for filters to clean the stream of noisy images. The conditions under which the observations were valid, as well as their cause were considered to customize the filter, thus providing a meaningful transformation of intensity data for the removal of undesired artefacts and restoration of the actual view. The evaluation of the filters involve the consistency between the images of the same view, the amount of distortion introduced where the filter is inappropriately used, the memory footprint, and most importantly, the execution speed in applying the filter.

A. *Range modification*

The difference in the average intensities of the three colour components across the whole image when observing a uniform grey scale object indicated that the proportional intensities between each pixel require adjustment for consistency. One non-random noise related contributor to this is the range limitation and offsets for each photo-sensors.

The calibration of data provided earlier indicates the possible range of the intensity data, which could be the result of compression or thresholding of the intensity scale. Using both the minimum and maximum intensities each pixel could detect, the scaling factor was determined and applied to reduce the difference in the intensities between each photo-sensor to lower the standard deviation score across the whole image when exposed to a uniform colour.

Although the effects were small due to the small minimum value, early results showed that the scaling of the range resulted in a higher standard deviation value, as shown in Table 1, thus prompting an alternative use of the minimum value to add to the intensity to determine if the variation was caused by a biased scaling for larger offset pixels. This experiment resulted in similar results where the standard deviation was larger, which suggests that the scaling based on the modified range introduced more randomness to the intensity data. Hence, this approach was abandoned and the range was left unmodified.

TABLE I. NOISE OFFSETS FOR RANGE MODIFICATION

	Original		Scaled	
	Ave	*StdDev*	*Ave*	*StdDev*
R	8.296	3.992	7.419	4.137
G	73.255	7.580	69.935	7.725
B	253.29	1.260	253.283	1.264
R	63.992	8.567	63.311	8.680
G	181.059	13.008	179.704	13.446
B	111.971	13.548	111.329	13.646
R	193.979	14.208	193.758	14.308
G	44.168	5.705	40.315	5.871
B	18.546	3.546	17.488	3.600

B. *Selective filter*

The by-product of blindly applying interpolation filters can cause the suppression of intended intensity characteristics, thus leading to the reduction in the ability to resolve key indicators such as sharp intensity changes and patterns which arise from a collection of small intensity changes. Sharpening algorithms, on the other hand, will enhance strong noise signals which require additional processing to detect and remove. This places an emphasis on correctly selecting the appropriate filter for the types of textures that are observed and anticipated. Rather than re-analyzing the image to correct the errors from filters, a selectively applied filter can avoid the lengthy decision process in identifying and correcting the incorrectly enhanced or suppressed textures.

One such selection approach is to observe the intensity relationship with its neighbours and use the difference in intensity to determine whether it requires any adjustment, or even be used as the coefficient to the weighting factor when deriving the new intensity. Additional information which can be integrated into the selection process is the intensity dependant noise characteristics of the camera which can identify the threshold values for the filter. A characteristic of random noise which was observed was such that the range of variation was limited and centered near the intended intensity of the object, thus the reverse lookup table for the original intensity with the corresponding probability can also be generated.

An example interpolation between two adjacent pixels if the intensity difference is below the standard deviation value could look like the following, where W represents the weighting, I_i and I_{i+1} represents the intensities at position i in the frame, and S_i represents the standard deviation at i, which is used as the threshold value in this example.

$$W = | I_i - I_{i+1} | / S_i \quad (9)$$

$$I_i = (I_i \,.(W + 1) + I_{i+1} \,.(1 - W)) / 2 \quad (10)$$

When analyzing the relative intensity with the neighbouring pixels, it is important to anticipate the increase in the processing load with respect to the number of neighbours it considers. For a rapidly changing ground texture being observed from close range, a large and frequent variation in intensity is often expected, thus the trends in the intensities are difficult to detect. This domain knowledge allows the search area for neighbour-based analysis to be kept very small.

Although the use of the calibration information to adjust the threshold and weighting factors are quite efficient and trivial to apply, the individual pixel and intensity based attributes consumes enormous amounts of memory to store the constants in lookup tables. Some of these relationships, such as the radial intensity shifts, can be modeled into a simple function to generate the values dynamically, but others, such as the pixel based standard deviation values must be stored individually. Compression of such lookup tables, or dynamically loading a sub-portion of the table using paging algorithms can be used to offset this, but it causes large spikes in the processing load, leading to delays and synchronization issues.

C. *Temporal filter*

A special kind of neighbour-based filtering involves the use of the time domain to analyze inter-frame trends in the intensity. Due to the limited fluctuation of the noise, averaging the intensity when the scene remains consistent results in a cleaner image. As mentioned earlier, the frame rate of the camera can determine the overall accuracy of the feature tracking algorithm, yet halving the theoretical frame rate to average the intensities with the previous frame allows significant reduction in random noise. One way to maintain the same frame rate and still allow inter-frame interpolation to occur is to keep track of the most recent frames in a circular buffer and interpolate between the frames within the buffer.

Like the other filter algorithms, blindly interpolating between multiple frames can introduce unwanted artefacts such as motion blur in this case. The larger the circular buffer size, the more consistent the intensity becomes, but the longer the corrupted frames for the motion blur is kept. A simple technique which uses a threshold to distinguish motion from noise can be implemented using the intensity difference or camera characteristics to isolate those pixels the intensity of which changes significantly. An individual pixel-based decision can be done quite efficiently, but this approach often causes scattering of the motion of the detected pixels, thus a constraint based regions or optical flow approaches can be used. Within the ground texture tracking domain, the motion detection constraint can be simplified greatly due to the assumption that the distance between the camera and the ground remains constant. This means the whole frame must move in sync, thus allowing a precise indicator of motion.

Although the potential for the temporal filter exists, the majority of images to be processed in this approach changes very rapidly. However, this filter can be used when detecting a longer life timed feature, such as landmarks.

D. *HSL colour scale*

An important characteristic with regard to how well the machine vision techniques mimic the human's visual perception techniques is in our ability to differentiate relative intensities far more accurately than absolute intensities. Absolute intensity is heavily dependant on many environmental attributes, but relative relationships between intensities often remain consistent in different types of environments, hence allowing a more generic framework for visual data analysis. A colour scale which allows a more natural comparison is the hue-saturation-luminance/value scale. This representation of colour allows alternative relationship measures between the pixel values by combining the red-green-blue sensors and allowing an additional trend to be made use of.

One common filter which can be applied with the use of the HSL colour scale is the shadow detector. Using the alternative colour scale and the hue/saturation characteristics, identifying the presence of shadows and suppressing the darkening in the intensity can be achieved by detecting a strong luminance change while the hue and saturation values stay within a small range which can be determined from the amount of noise anticipated.

In the context of ground texture analysis, the presence of shadows are unlikely with correct precautions and the variation in the hue or saturation measures can be quite small, but the alternative distance measurement and the ability to measure a more ambient condition independent value can add to the confidence in detecting useful landmarks.

The conversion process between the RGB colour scale and the HSL involves a simple mapping process after determining the minimum and maximum of the three intensities and determining which sector the hue is in. Hence a dynamic conversion between the two colour scales can be done easily without the need for a lookup table.

E. *Quantization blocks*

A close inspection of the images acquired from the webcams indicated that the images had undergone a lossy compression algorithm during the encoding phase to allow large streams of data to be transmitted. This introduced artefacts into the image, causing bias in the behaviour of certain image processing algorithms. The behaviour indicated a similar pattern to the quantization phase used in JPEG compression algorithms, but with a smaller block size of 4 by 4 pixels, and a weaker block for the inner 2 by 2 pixels. Using a higher resolution to capture the image, then manually interpolating the neighbouring intensities did not show much improvement, as the effect of the quantization was stronger at higher resolutions as the throughput of the data transmission needed to be maintained. To compensate for this block formation, the pixels on the borders of the blocks were interpolated with their neighbours on the other side of the border. The effects and the weighting can be seen in Fig. 6.

Figure 6. Quantisation blocks: original, 3x3 gaussian blur, quantisation recovery filter, weighting mask

V. SUMMARY

The use of a webcam to track ground feature displacement allows an accurate localization technique with capabilities for operation in many different types of environments for vehicles moving on a solid surface.

The various conditions and configuration issues of the webcam, as well as the analysis of the image filter algorithms to restore the image from the noise prone sensor were discussed. The findings indicate that although many characteristics can be found to improve the quality of the images, the small amount of noise reduction in most filters does not affect the ability to track the ground texture motion. This is due to the assumptions and constraints that are placed on the images to assist in efficiently extracting the required information from the image, as well as the detailed texture of the ground.

Quantification of the increased image quality is difficult to measure, thus the comparison was done by visual inspection to observe the amount of change in the pixel intensities around feature rich and noisy areas of the image. The performance measure was also conducted to indicate feasibility for real time operation while minimizing memory usage.

The observed results indicate that the standard deviation for various levels of luminance allowed reasonable indication for differentiating between intended and noise generated intensity changes. To minimize the memory usage, the trend was modeled by a polynomial function between the standard deviation and the intensity value. Another useful model which was integrated was the radial shift offset, as the effect of the intensity change around the image was quite significant especially in the absence of variety in the image intensities, such as flat table tops and vinyl floors. The last filter which was determined to be well suited was the quantization block suppression filter. Although the weighting coefficients used in the interpolation were not fully explored, the improvement in image quality was very obvious.

The application of the discussed settings and algorithms are currently being examined in conjunction with feature detection and tracking algorithms with initial results showing promising ability to track the robot displacement with sub-millimeter precision with minimal errors being encountered. Other areas of potential application of this texture matching technology are in stereo disparity mapping and super-resolution image processing.

REFERENCES

[1] G. Dudek and M. Jenkin, Computational Principles of Mobile Robotics, Cambridge University Press, 2000.

[2] S. A. Hutchinson, G. D. Hager, and P. I. Corke. “A tutorial on visual servo control”, IEEE Trans. Robot. Automat., 12(5):651--670, Oct. 1996.

[3] S. Krootjohn, “Video image processing using MPEG Technology for a mobile robot,” PhD Thesis, Vanderbilt University, 2007.

[4] T.W. Ng, “The optical mouse as a two-dimensional displacement sensor”, Sensors and Actuators A, 107, 2003.

[5] T. Pavlidis, Algorithms for Graphics and Image Processing, Springer-Verlag, Berlin, 1982.