14th National Conference on Machines and Mechanisms (NaCoMM09),
NIT, Durgapur, India, December 17-18, 2009 **NaCoMM-2009-RJB3**

The Generalized Feature Vector (GFV) : A New Approach for Vision Based Navigation of Outdoor Mobile Robot.

J. Bhattacharya [a] and S. Majumder [b*]
[a,b] CMERI, Durgapur, WestBengal, India

* Corresponding author (email: [b] sjm@cmeri.res.in)

Abstract

This paper studies various methods of feature detection and introduces a new approach termed as Generalized Feature Vector which essentially encapsulates multiple feature parameters for consistent detection. The basic idea of GFV stems out from the fact that if detection relies on a single feature it may lead to multiple false alarms or wrong identification. This error can be greatly reduced to a large extent when a number of features are used instead of one. Though primarily applied to vision sensors GFV has the potential to include data obtained from other types of sensors. Experimental analysis of the proposed method is also included, which shows that the approach is resilient to the extrinsic parametric variations with minimal false alarm rate. This also shows that GFV method has tremendous potential in areas like robot navigation, surveillance, remote sensing and many more.

Keywords: Computer vision, feature detection, color matching

1.Introduction

Detection of feature from exteroceptive sensors has remained an important area of research for several reasons. Firstly it provides the unique opportunity to abstract and encapsulate the dominant and distinguishable characteristics of the environment or scene from the sensory data. Secondly it is a process of reducing the resource requirement and the associated complexity of handling large data sets in real-time. Often features are defined as geometric primitives such as point, line, arc segments or some form of derived entities such as color and texture for example. In general, features segregate “objects of interest” from the raw sensory data.

Many methods have been proposed by various researchers for consistent feature detection. A few among these are histogram color detection [1,2 and 3], texture detection based on Wavelet transforms such as Daubechies wavelets, Haar Wavelets, Morlet wavelets [4,5,6,7,8 and 9], edge and corner detection [10,11,12,13 and 14], and optical flow detection [15] etc. Frame differencing methods have been applied for detecting moving objects using single or multimodal statistical Gaussian models [16 and 17]. Besides, template matching has always been used as a primary tool by many computer vision researchers. Fractal and phase spectrum methods have also been used for feature detection [18,19 and 20].

Inspite of many uniqueness and advantages, most of these methods require large computational power and hence unsuitable for real-time navigation and tracking application. Another limitation is that many of these methods have been developed for a specific and well-structured indoor environment for specific applications and consider camera calibration as a prerequisite rendering themselves unsuitable for outdoor and unstructured environment where extrinsic parameters are dominant rather than intrinsic.

This paper is organized in the following manner. Section 1 provides basic background of the problem. This section also includes an outline of various significant work carried out for consistent feature detection. Section 2 defines the GFV framework and its comparison with other conventional approaches. This section also includes the algorithms developed and its application to experimental dataset. Section 3 deals with results and discussions on experimental findings, where as Section 4 provides an in-depth analysis of the performance of the present algorithm. Finally, conclusion of this work is presented in Section 5.

2. Definition of GFV

Present work aims at detecting features and representing them in a generalized feature vector (GFV), which can be used to uniquely identify each of the dominant objects in an image. The basic idea of using GFV as a scene descriptor stems out of the fact that point features often require a secondary level of corroboration such as color and texture to make it invariant. In principle GFV can include as many parameters as desired. The concept of GFV and its parametric space is further elaborated in Fig. (1) & (2) respectively. Another uniqueness of GFV is that it can also accommodate “feature parameters obtained from other co-located sensors”. There is no limit on how many feature parameters can be included in GFV. Although inclusion of multiple parameters can improve the detection reliability it however may increase the computation cost. It has also been observed (during computation) that for optimal performance not more than three parameters should be used. However the actual number of parameters will depend on the application requirements and available computational resources. The generalized feature vector (GFV) used in the work reported here is considered to be a multidimensional entity which can include multiple parameters like color, shape, energy, entropy, skewness, size ratios and many more. Some of these parameters may be orthogonal to the other. This also enables tracking of a similar object from a sequence of images taken at different time or depicting different environments making it immuned to viewpoint and environmental changes.

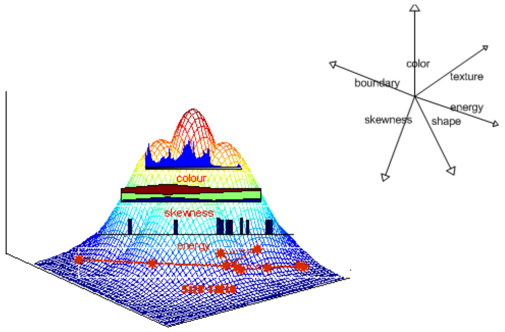


Fig-1: In statistical sense the GFV can be described as a multimodal probability distribution in a multidimensional feature space depicted above.

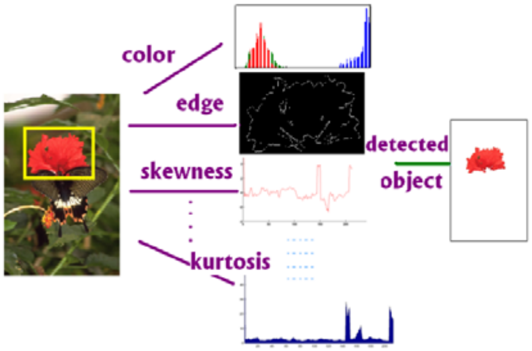
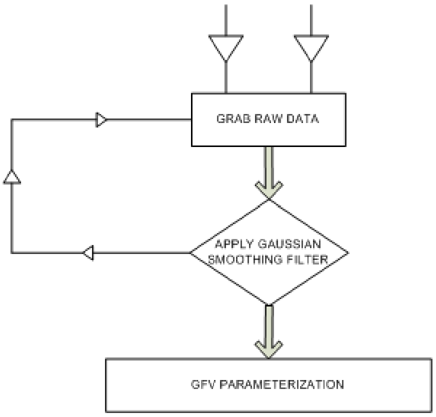


Fig-2: Various features such as color, boundary, texture, skewness, energy etc are combined to create the generalized feature vector as shown above[1].

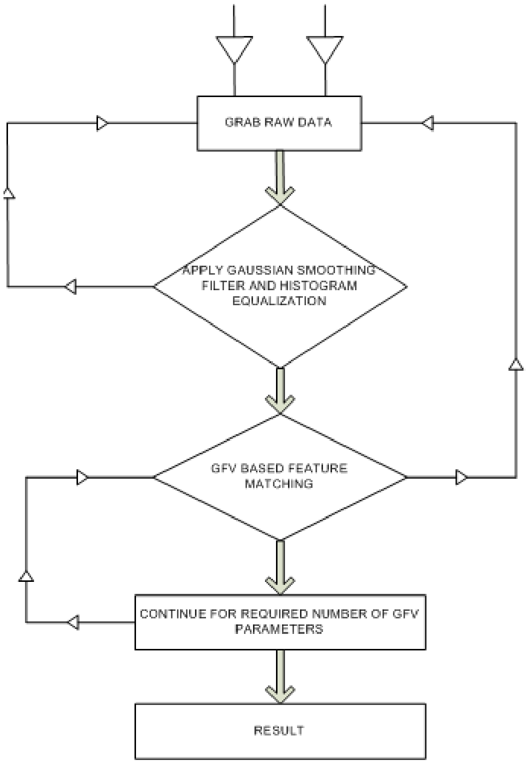
2.1 The GFV Algorithm

The algorithmic flow of the present method may be described in the following manner:

During first step a reference model of GFV is created which is then applied to the actual data. This approach is further depicted in Fig. (3) in the form of flow diagram.



Step-1: Reference model creation:



Step-2: Application of GFV model to experimental data

Fig-3: Block diagram for the present method

2.1.1 Step-1: Reference model creation:

1. A Gaussian filter of unity σ with a window size of 7 pixel width is used for smoothing and noise reduction.

---

[1] Feature parameters used in this work is elaborated in Section 2.

2. Object of interest is manually selected from image data using a selection window *'w'*, and its feature vector is created. The parameters of the feature vector for this particular case include color, mean, standard deviation, shape, energy, entropy, skewness and size ratios. Mathematical approaches used for computation of some of these parameters are shown below:

(i) Parameter: Color

A probability density function (*pdf*) of the selected object I^{rgb}[2] is calculated for three primary colors red, green and blue. The *pdf* for each channel red, green and blue represented by p_i^r, p_i^g and p_i^b is calculated in the following manner:

$$p_i^r = \frac{I_i^r}{\sum_{i=0}^{N} I_i^r} \tag{1}$$

$$p_i^g = \frac{I_i^g}{\sum_{i=0}^{N} I_i^g} \tag{2}$$

$$p_i^b = \frac{I_i^b}{\sum_{i=0}^{N} I_i^b} \tag{3}$$

where i represents the bin number ranging from *0-N*, I_i^c represents the number of pixels in the i^{th} bin of the c^{th} *(c ε{red(r),green(g),blue(b)})*color channel. Intensity range L^c-H^c corresponding to the *pdf* value above a threshold of p_{thresh} (which is half the maximum *pdf* value for that particular channel) is selected as the color intensity representing the object.

(ii) Parameter: Statistical Mean

The mean of the intensity range L^c-H^c selected in the previous part for each color channel is calculated separately using Eq. (4), (5) and (6).

$$\mu^r = \frac{\sum_{i=L^r}^{H^r} bin_i^r}{|L^r : H^r|}, \tag{4}$$

$$\mu^g = \frac{\sum_{i=L^g}^{H^g} bin_i^g}{|L^g : H^g|}, \tag{5}$$

$$\mu^b = \frac{\sum_{i=L^b}^{H^b} bin_i^b}{|L^b : H^b|} \tag{6}$$

Here bin_i^c represents the i^{th} bin intensity of the c^{th} color channel of RGB image I^{rgb}, whereas bin_i represents the i^{th} bin intensity of grayscale image *I*.

(iii) Parameter: Statistical Standard Deviation

The standard deviation (of intensity range L^c-H^c) of red, green and blue color channel can be calculated using Eq. (7) - (9) and represented by σ_r, σ_g and σ_b respectively.

$$\sigma_r = \sqrt{\sum_{i=L^r}^{H^r} (bin_i^r - \mu^r)^2 \, p_i^r} \tag{7}$$

$$\sigma_g = \sqrt{\sum_{i=L^g}^{H^g} (bin_i^g - \mu^g)^2 \, p_i^g} \tag{8}$$

$$\sigma_b = \sqrt{\sum_{i=L^b}^{H^b} (bin_i^b - \mu^b)^2 \, p_i^b} \tag{9}$$

In a similar manner parameters like skewness (S_w), energy (e_w), and entropy (Φ_w) are given by Eq. (10), (11) and (12) respectively. These three parameters are calculated using the grayscale image '*I*' of the selected object.

(iv) Parameter: Skewness

$$S_w = \frac{1}{\sigma^3} \sum_{i=0}^{N} (bin_i - \mu_m)^3 \, p_i \tag{10}$$

where μ_m and σ are the mean intensity and standard deviation of the grayscale image *I* respectively and p_i represents the *pdf* of the i^{th} bin of image *I*.

(v) Parameter : Energy

$$e_w = \sum_{i=0}^{N} p_i^{\,2} \tag{11}$$

(vi) Parameter : Entropy

$$\Phi_w = \sum_{i=0}^{N} p_i \log_2 p_i \tag{12}$$

Standard definition of energy and entropy terms with reference to computer vision is applied here [21].

---

[2] For details refer to Annexure-1

2.1.2 Step-2: Application of GFV to experimental data.

1.Gaussian smoothing and image enhancement techniques are used to improve each image *IM* of the experimental set as a general preprocessing step.
2. A 2 σ or 3 σ variation δ_s of the intensity ranges selected in section 2.1.1.2(i) is used to filter each image. This is done in two steps using Eq. (13) and (14) :

Initially the outlying range (those which do not lie in the range L^c-H^c selected in section 2.1.1.2(i)) of intensities for each color channel is calculated using Eq. (13).

$$O_c = (abs((IR - m_c) > (\varsigma_s . \sigma_c)))$$

$$C \in \{red, green, blue\}, IR = 0:255 \qquad (13)$$

The regions representing same color of the selected object can now be determined from Eq. (14) given below.

$$S(reg) = ((IM_r -- \neg Or) \wedge (IM_g -- \neg Og) \wedge (IM_b -- \neg Ob)) \qquad (14)$$

where, IM_r, IM_g and IM_b denote the intensities of the red, green and blue color channels respectively. O_r, O_g and O_b denote the detected intensity outliers of the three color channels. '$\wedge$' and '$\neg$' used in the equation above are *'logical AND'* and *'logical NOT'* operators respectively.
3. The feature parameters (mean, standard deviation, skewness, energy etc) are now calculated for each of the regions generated in the previous step.
4. The final step is to match the calculated feature vector of each of the regions with the vector of the reference model (created in section 2.1.1) to find the best matching region. The search space consists of the parameter set $s = (\Phi c$[3]$, \Phi s, \Phi m, \Phi_{\sigma}, \Phi e_w, \Phi_{\Phi w}, \Phi_{Sw}, \Phi sr$[4]$)$ where Φc, Φs, Φm, Φ_{σ}, Φe_w, $\Phi_{\Phi w}$, Φ_{Sw}, Φsr denote the matching percentage of each of the parameters color, shape, mean, standard deviation, energy, entropy, skewness, size ratios respectively for each identified region to be matched against the predetermined feature vector. The best matching region is now selected using Eq. (15).

$$s^* = \arg\max(\Phi c(i) + \Phi s(i) + \Phi m(i) + \ldots + \Phi sr(i)) \qquad (15)$$

Code level details regarding the implementations of the algorithm could not be provided due to space constraints. However, the algorithm shown here has been developed using references [22 and 23.]

3. Results and Discussions

Color intensity of the object is selected in two ways[5]. The following test image in Fig. (4) is first used to evaluate the performance of each method of intensity selection for the object to decide which method is to be followed for feature detection from any random image sequence for better performance.

Fig-4: The red object highlighted in the figure using a yellow rectangle depicts the selected object.

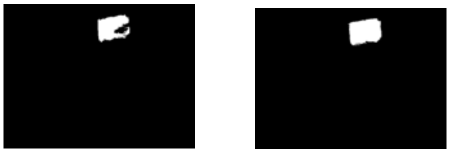


Fig-5: (a) output image (b) output image using *3σ*
range using *2σ* range

Fig-6: *2σ* range filtered image

Initially range selection is done using dominant color *pdf* plot only. The highlighted object as shown in Fig. (4) is detected from the image as shown in Fig. 5(a) and (b) using a *2σ* and *3σ* filter respectively.

Range selection done using all three color *pdf* plot gives the filtered image with a *2σ* range filter as seen in Fig. (6).

Table 1. shows the intensity ranges calculated using *pdf* plot of red channel (dominant color for the sample chosen above.) The corresponding green, blue values of pixels lying under selected red intensity range is shown below. Table 2. depicts the intensity ranges when calculated from the *pdf* plot of all the three color channels. From the tables shown below, it can be observed that a better result is obtained when the color intensities are selected using *pdf* plots of all the three colors red, green

---

[3,4,5] For details refer to Annexure-1

and blue. The filter gets narrower when only dominant color is used for selection and thus the entire region is not properly extracted.

TABLE -1.Intensity ranges calculated using dominant color *pdf* plot only

Color channels	red pdf plots	*2σ* filter	*3σ* filter
red	210-255	194-255	194-255
Corresponding green	75-131	84-136	68-148
Corresponding blue	80-136	86-138	70-149

TABLE -2: Intensity ranges calculated using all three color *pdf* plots

Color channels	From *pdf* plots	By using *2σ* filter
Red	210-255	194-255
Green	60-124	37-149
Blue	70-120	52-140

Experimental analysis has been carried out from a set of indoor and outdoor image sequence. Some of the results obtained are shown below.

(i) Indoor Experiment I: Geometrically same objects with different color

Fig-7: Geometrically same objects are shown in (a) and detected objects using GFV is shown in (b)

It can be seen that when any one red chair is selected as the object of interest the output image displays both the red regions as all their feature parameters are same. The blue region however doesn't get selected as its color parameter is different though all the rest features are equivalent.

(ii) Indoor Experiment II: Geometrically different objects with nearly same color.

Inspite of having very minute color variations, only the object selected is detected in each frame. The algorithm is thus found to possess the ability to differentiate even very minute color variations, making the detection robust and less error prone.

Fig-8: Geometrically different objects with nearly same color can also be detected.

(iii) Outdoor experiment

Fig-9: Red colored object is detected as shown in the inset figure at the top right corner

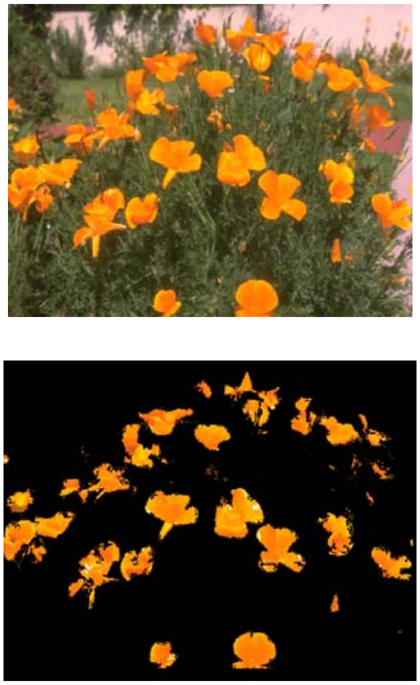


Fig-10: Objects of a specified color (as shown in the bottom picture) can also be detected very accurately rejecting *out lier* data

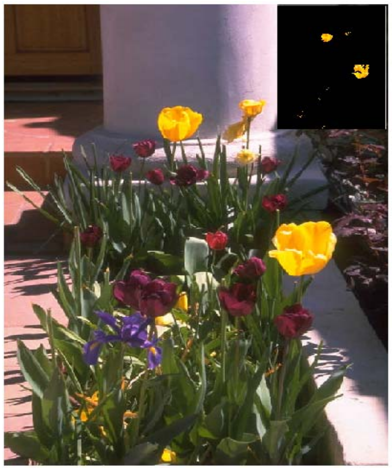


Fig-11: Yellow flowers are detected as shown in the inset figure at the top right

4.Performance

It is thus observed that GFV method can detect object reliably with negligible false alarm or wrong identification. The performance and evaluation of the system is found to be very stable. An object can be detected in a single frame in 172 *ms* when only color is used as a parameter. On involving shape detection, size ratios, skew along with the color, the time taken is 810 *ms*. Minimal processing time of the same algorithm when applied to a sequence of 50 frames is 7.87 *sec*. It has also been observed that variation in computational performance is not significant when number of GFV parameters were increased for a certain parameter class. This means when four size ratios such as ratios between convex area and area, perimeter of minimum bounding rectangle and perimeter of the object, foci and major-axis length of the enclosing ellipsoid, distance of two extrema points (top-left, bottom-right) from the centre is used for matching instead of two size ratios such as eccentricity, distance of two extrema points (top-left, bottom-right) from the centre; it shows a marginal change of 70 *ms*. In a similar manner when three intensity parameters (e.g. skewness, energy, entropy) is used instead of one (e.g. energy) ; it exhibits an increase of computational time by 50 *ms*. This performance has been obtained without optimizing the code on an Intel Pentium 4 machine, with CPU *3GHz* and Memory *1GB* using Matlab *7.1*.It is obvious that this performance will increase considerably when deployed using Matrox Morphis Frame grabber card with built in DSP chipset.

5.Conclusion

In general any robust feature detection algorithm should resolve consistently the following issues: 1.View point invariance, 2. Occlusion (including shadow and silhouette), 3. Scale, 4. Resource requirement (CPU speed & memory) . For any real-time application the feature detection method should exhibit resilience against these issues. The main limitations of color matching algorithm reported in literature are that most of them are unable to cope with the changes of intensity variations and are directly influenced by the extrinsic factors. Preprocessing and calibration methods are incapable to solve these problems consistently and reliably in addition to the problem of rejecting *outlier* data. The proposed GFV method has been found to be capable of resolving these issues and can track object from a sequence of images taken at different time or different environment making it immune to the problem of view point and environmental changes.

Work is in progress to implement GFV approach for vision based Mobile Robot navigation at CMERI for outdoor environment. Other future applications may include surveillance, remote sensing, gating analysis, traffic management and many more.

REFERENCES

[1]. "Likelihood based object detection and object tracking using color histograms" , Withagen, Klamer Schutte1and Frans Groen, Proc. ICIP2002, September 22-25, 2002, Rochester, New York

[2]. “Polygon feature extraction from Satellite imagery based on color image segmentation and medial axis”, Ojaswa Sharmay, Darka Miocz and Franc¸ois Antony, The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences. Vol. XXXVII. Part B3a. Beijing 2008

[3]. “Frontal-view face detection and facial feature extraction using color, shape and symmetry based cost functions” , Eli Saber , A. Murat Tekalp, 1998 Elsevier Science B.V.

[4]. “Texture Features for Browsing and Retrieval of Image Data”,B.S. Manjunathi and W.Y. Ma, IEEE

[5]. **“**Extracting Texture Features from Arbitrary-shaped Regions for Image Retrieval”,Ying Liu Xiaofang Zhou Wei-Ying Ma

[6]. “Brushlet Features for Texture Image Retrieval” , Chibiao Chen and Kap Luk Chan, Chi-Fa Chen, DICTA2002

[7]. “Enhancing capabilities of Texture Extraction for Color Image Retrieval”, Pranam Janney, Sridhar G, Sridhar V., PWASET VOLUME 5 APRIL 2005

[8]. “Image Classification Based on Textural Features using Artificial Neural Network” ,Prof S K Shah, V Gandhi

[9]. “Multichannel Texture Analysis Using Localized Spatial Filters “, Alan Conrad Bovik, Marianna Clark, And Wilson S. Geisler, IEEE

[10]. “Object Tracking using Color Interest Points” ,P. Gabriel, J.-B. Hayet, J. Piater, and J. Verly, IEEE

[11]. “Points of interest extraction using pairwise clustering and spatial features”, Dimitrios Besiris, Nikolaos A. Laskaris, E. Zigouris, George Economou and Spiros Fotopoulos

[12]. “Visual mapping by a robot rover” , Moravec, H. 1979, In Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI). 598–600.

[13]. “Good features to track”, Shi, J. And Tomasi, C. 1994,In IEEE Conference on Computer Vision and Pattern Recognition (CVPR). 593–600.

[14]. “Space-time tracking”, Torresani, L. And Bregler, C. 2002, In European Conference on Computer Vision (ECCV). 801–812

[15]. “Statistical shape knowledge in variational motion segmentation”, Cremers, D. And Schnorr, C. 2003. ,ISrael Nent. Cap. J. 21, 77–86.

[16]. “Real-time tracking of the human body”, Wren, C., Azarbayejani, A., And Pentland, A. 1997. Pfinder IEEE Trans. Patt. Analy. Mach. Intell. 19, 7, 780–785.

[17]. “Background and foreground modeling using nonparametric kernel density estimation for visual surveillance”, Elgammal, A.,Duraiswami, R.,Harwood, D., Anddavis, L. 2002, Proceedings of IEEE 90, 7, 1151–1163.

[18]. “Image Feature Extraction and Segmentation using Fractal Dimension”,Yuxin Liu, Yanda Li, IEEE

[19]. “Image Feature Extraction Based on Phase Spectrum”,Yaoxin LV. Zlu-Qiang Liu. Xiangtua Zhu, IEEE

[20]. “Image Feature Extraction Based on The Two-Dimensional Empirical Mode Decomposition” , Wan Jian , Ren Longtao, Zhao Chunhui, IEEE

[21]. “Color Histogram Features Based Image Classification in Content-Based Image Retrieval Systems" , Szabolcs Sergy´an , IEEE.

[22]. Matlab Image Processing Toolbox, http://www.mathworks.com/access/helpdesk/help/toolbox/images/

[23]. Matrox Imaging Library for Image Acquisition and Processing, http://matrox.com/imaging

ANNEXURE I

2. In this paper the terms I^r , I^g , I^b represents the intensity values of each red, green and blue respectively, whereas I denotes grayscale values of the RGB Image I^{rgb}.

3. Φc is determined from histogram matching of each region in S and the selected object.

4. $\Phi sr = {}^{1}\!/_{n} \sum_{i=1}^{n} \Phi\mathrm{sr}(i)$ where Φsr denotes the mean of all the size ratios.

5.The intensity ranges can be selected in two ways. The *pdf* of the object of interest for all the three color channels may be used as described in section 2.1.1.1(i). Alternatively the *pdf* of the dominant color range may be used for that color similarly.