ISSN 2278-3091

International Journal of Advanced Trends in Computer Science and Engineering, Vol.2 , No.1, Pages : 171-177 (2013)
Special Issue of ICACSE 2013 - Held on 7-8 January, 2013 in Lords Institute of Engineering and Technology, Hyderabad

COMPARISION OF REGRESSION WITH NEURAL NETWORK MODEL FOR THE VARIATION OF VANISHING POINT WITH VIEW ANGLE IN DEPTH ESTIMATION WITH VARYING BRIGHTNESS

Murali S[1],Deepu R[2],Vikram Raju[3]

Department of Computer Science and Engineering
Maharaja Institute of Technology, Mysore, Karnataka, India
[1]murali@mitmysore.in, [2]rdeepusingh@mitmysore.in,[3]vikramraju17@gmail.com

Abstract

A vanishing point in a 2D image is a point where parallel edges in 3D scene converges due to perspective view. Vanishing point provides lot of information in most of the applications like distance of objects in 3D Scene. This point depends on Camera viewing angle as well as distance between the parallel edges in a scene. In order to find relationship between the Vanishing Points with view angle and distance between parallel edges with varying focal lengths of the camera, various data were collected by having an experimental set-up by varying the distance between two parallel sticks in an image and also by varying the angle of the camera with the parallel sticks. Images are preprocessed to overcome the problems of improper lighting in the scene. Different data mining techniques were used and the results are twofold. The results of regression model for the relationship between the angle of the camera and the X and Y coordinates of the vanishing point has been analyzed using neural network model, which gave 100% validity on the model.

Keywords

Vanishing Point, Regression Model, ANOVA, Analysis of Variance, Image Analysis, Angle Detection, Neural Networks, Multilayer Perceptron.

Introduction

Vanishing points are useful in applications such as road detection in unmanned cars, object detection in an image and training canines etc. In automated vehicles driving on the roads, it can be used to monitor and control the vehicle to be kept in the specified lane of the road when it deviates by tracing the vanishing point. It can also be used in embedded system applications to assist the visually impaired to direct them on a pathway. The method to find the vanishing point works with simple 8-bit gray scale image. The fundamental visual element is the trend line or "directionality" of the image data, in the neighborhood of a pixel. This directionality is measured as the magnitude of the convolution of a pixel's neighborhood with a Gabor filter. Gabor filters are directional wavelet-type filters, or masks. They consist of a plane wave, in any direction in the image plane, modulated by a Gaussian around a point. The Gaussian localizes the response and guarantees that the convolution integral converges, and the plane wave affords a non-zero convolution over discontinuities that are perpendicular to the wave's direction. In other directions, the periodicity of the wave brings the convolution close to zero. The phase of the wave, relative to a feature at the origin, is best accounted for by a sine/cosine combination of filters: cosine responds to even-parity features, sine responds to odd-parity features, and the other responses are vector combinations of these two. We are interested only in the magnitude of the response. The common formulas for the Gabor filter are:

$$G_{cos}(x, y) = e^{\frac{\left(-x'^2 + \gamma^2 y'^2\right)}{2\sigma^2}} \cos(\omega x')$$

$$G_{sin}(x, y) = e^{\frac{\left(-x'^2 + \gamma^2 y'^2\right)}{2\sigma^2}} \sin(\omega x')$$

where the wave propagates along the x' coordinate, which is rotated in some arbitrary direction relative to the image coordinates x and y; ω is the circular frequency of the wave, $\omega = 2\pi / \lambda$. Parameter σ is the half-width of the Gaussian, and defines roughly the size scale on which discontinuities are detected. Parameter γ is the eccentricity of the Gaussian: if $\gamma \neq 1$, the Gaussian is elongated in the direction of the wave, or perpendicular to it. Repeated convolutions of the same mask with every pixel's neighborhood can be accelerated by convolving in the Fourier space, where the convolution of two functions is expressed as the product of their Fourier transforms.

Intuitively, one can think of the same mask being translated across the image. In Fourier space, translations become multiplications by a phase factor, and so it is possible to transform the image and the mask once, multiply the transforms, then perform the inverse transform to obtain the value of the convolution at every pixel of the image at once. In the program used, the image is convolved with Gabor filters in a (large) number of evenly spaced directions, from zero to 180 degrees, and the direction with the strongest response is retained as the local direction field (trend line) of the image. Naturally, this calculation is by far the most computationally intensive part of the algorithm [1]. A well-known Fast Fourier Transform library FFTW is used.

Researchers have proposed various methods to determine the vanishing point of 2D images. Wolfgang Forstner[5] presented a method where in the vanishing points were estimated from line segments and their rotation matrix, using spherically normalized homogenous coordinates

ISSN 2278-3091

International Journal of Advanced Trends in Computer Science and Engineering, Vol.2 , No.1, Pages : 171-177 (2013)
Special Issue of ICACSE 2013 - Held on 7-8 January, 2013 in Lords Institute of Engineering and Technology, Hyderabad

and allowing vanishing points to be at infinity. The proposed method had minimal representation for uncertainty of homogenous coordinates of 2D points and 2D lines and rotations to avoid use of singular covariance matrices of observed line segments. A more efficient method was proposed by Danko Antolovic, Alex Leykin, and Steven D. Johnson, where line segments were filtered using Gabor filters and having a convolution close to zero. The output of Gabor filters were processed to Fast Fourier Transformations. There are many methods to determine the vanishing point, in this study, a model will be formed trying to bring out a relationship between the angle of camera and the position of the vanishing points.

Methodology

Numerous photographs were taken by altering the distance between the sticks and the angle between the camera and the horizontal. Here the angle (α) gives the angle between the camera axis and the horizontal. Using the vanishing point method described by Danko Antolovic, Alex Leykin, and Steven D. Johnson, the vanishing point was identified for the various data set images. The camera angle α was varied from 60^o to 120^o with increments of 10^o, for every alteration in the distance between the sticks from 25cms to 115cms apart with increment of 10cms for each trial. The images were taken for varying intensities for efficient vanishing point detection. The resultant data was used to build a relationship between them. A schematic of the experiment is shown below:

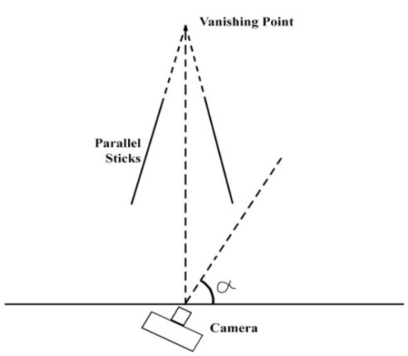

Figure 1: Schematic of the experimental set-up.

Phase – I

General Regression Analysis

A general regression analysis is carried out for the data set for various angles (α) and the X and Y coordinates of the vanishing point determined for the various images. The following table shows the initial results.

Table 1: Regression Coefficients

Term	Coefficient	SE Coefficient	T	P
Constant	231.480	14.8529	15.5848	0.0
X	-0.056	0.0122	-4.5650	0.0
Y	-0.369	0.0369	-10.0217	0.0

From the above table we get the regression coefficients for the obtained regression model. The model equation is defined by,

$$\alpha = 231.480 - 0.056X - 0.369Y$$

Here, by obtaining the X and Y coordinates of the vanishing point, the camera angle α can be determined. The corresponding P value for the constant, X and Y coordinates is 0, thus having a strong significance in the model.

Phase – II

Neural Networks and Multilayer Perceptron

A Multilayer Perceptron (MLP) is a feed forward artificial neural network model that maps sets of input data onto a set of appropriate output. MLP utilizes a supervised learning technique called back propagation for training the network. MLP is a modification of the standard linear perceptron and can distinguish data that is not linearly separable.

Input Layer - a vector of predictor variable values (X_1 . . . X_p) is presented to the input layer. The input layer standardizes these values so that the range of each variable is -1 to 1, using the equation mentioned below. The input layer distributes the values to each of the neurons in the hidden layer. In addition to the predictor variables, there is a constant input of 1.0, called the bias that is fed to each of the hidden layers, this bias is multiplied by a weight and added to the sum going into the neuron.

Hidden Layer - arriving at a neuron in the hidden layer, the value from each input neuron is multiplied by a weight (W_{ji}), and the resulting weighted values are added together producing a combined value U_j. The weighted sum (U_j) is fed into a transfer function, σ, which outputs a value H_j. The outputs from the hidden layer are distributed to the output layer. In this case since all data are equally important, all have same weights.

Output Layer-arriving at a neuron in the output layer, the value from each hidden layer neuron is multiplied by a weight (W_{kj}), and the resulting weighted values are added together producing a combined value V_j. The weighted sum (V_j) is fed into a transfer function, σ, which outputs a value Y_k. The Y values are the outputs of the network.

Missing value handling for this experiment consists of user and system-missing values are treated as missing. Statistics are based on cases with valid data for all variables used by the procedure. Weight handling is not applicable in this experiment, since all coordinates and angles and distance between sticks are equally important and significant to the study.

Multilayer Perceptron: Angleαvs. X Coordinate

147 of 210 observations were used to build the models which were later used to predict the remaining observations.

ISSN 2278-3091

International Journal of Advanced Trends in Computer Science and Engineering, Vol.2 , No.1, Pages : 171-177 (2013)
Special Issue of ICACSE 2013 - Held on 7-8 January, 2013 in Lords Institute of Engineering and Technology, Hyderabad

Table 2: Case Processing Summary

		N	Percent
Sample	**Training**	147	70.0%
	Testing	63	30%
Valid		210	100.0%
Excluded		0	
Total		210	

The input layer consists of two covariates, the distance between the sticks in the experiment and the X coordinate of the vanishing point. It thus consists of two units, excluding the bias unit in the set-up. The rescaling method used for the covariates was standardizing. The dependent and independent variable were standardized using,

$$\frac{X - Mean}{Sample\ Size}$$

Only one hidden layer is used with three units in the hidden layer excluding the bias unit. The activation function used was the hyperbolic tangent function.

The output layer depends only on the X coordinates with only one unit in it. A standardized rescaling method was used for the scale dependents. An identity activation function was used in this case. To incorporate the error, a sum of squares error function was incorporated.

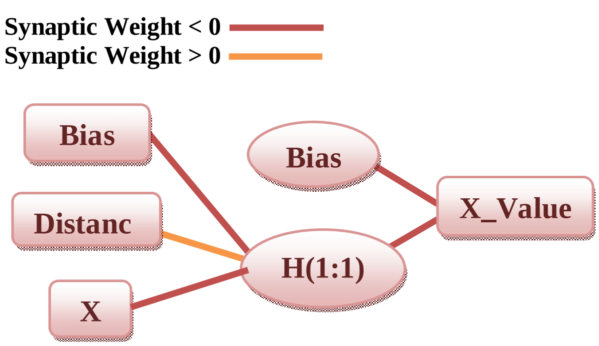

Figure 2: Multilayer Perceptron model for Angle α vs. X Coordinate

Table 3: Model Summary

Training	Sum of Squares Error	33.206
	Relative Error	0.730
	Stopping Rule Used	One consecutive step(s) with no decrease in error (error computations are based on the testing sample)
Testing	Sum of Squares Error	8.478
	Relative Error	0.510

SSE is the error function that the network tries to minimize during training. Note that the sums of squares and all following error values are computed for the rescaled values of the dependent variables. The relative error for each scale-dependent variable is the ratio of the sum-of-squares error for the dependent variable to the sum-of-squares error for the "null" model, in which the mean value of the dependent variable is used as the predicted value for each case.

Table 4: Parameter Estimates

Predictor		**Predicted**	
		Hidden Layer 1	**Output Layer**
		H(1:1)	X Coordinates
Input Layer	(Bias)	-0.110	
	Distance	0.038	
	X	-0.821	
Hidden Layer 1	(Bias)		-0.137
	H(1:1)		-0.928

Table 5: Independent Variable Importance

	Importance	**Normalized Importance**
Distance	0.040	4.2%
X	0.960	100.0%

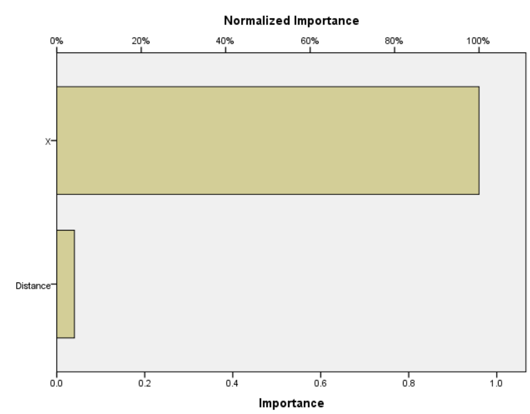

Figure 3: Variable Importance

Summation of importance is 1. Hence the importance of each of the independent variables in predicting the dependent can be understood. From the figure we can clearly understand the importance of the X coordinates of the Vanishing point for various camera angles α, and has a 100% importance level, while the distance between the sticks do not account much for the model. This agrees with the first regression equation initially derived.

Multilayer Perceptron: Angle α vs. Y Coordinate
139 of 210 observations were used to build the model which was later used to predict 71 of the remaining observations.

ISSN 2278-3091

International Journal of Advanced Trends in Computer Science and Engineering, Vol.2 , No.1, Pages : 171-177 (2013)
Special Issue of ICACSE 2013 - Held on 7-8 January, 2013 in Lords Institute of Engineering and Technology, Hyderabad

Table 6: Case Processing Summary

		N	Percentage
Sample	**Training**	139	66.1
	Testing	71	33.8
Valid		210	100.0
Excluded		0	
Total		210	

The input layer consists of two covariates, the distance between the sticks in the experiment and the Y coordinate of the vanishing point. It thus consists of two units, excluding the bias unit in the set-up. The rescaling method used for the covariates was standardizing. The dependent and independent variable were standardized using,

$$\frac{\boldsymbol{Y - Mean}}{\boldsymbol{Sample\ Size}}$$

Only one hidden layer was used with three units in the hidden layer excluding the bias unit. The activation function used was the hyperbolic tangent function.

The output layer depended only on the Y coordinates with only one unit in it. A standardized rescaling method was used for the scale dependents. An identity activation function was used in this case. To incorporate the error, a sum of squares error function was incorporated.

Synaptic Weight < 0
Synaptic Weight > 0

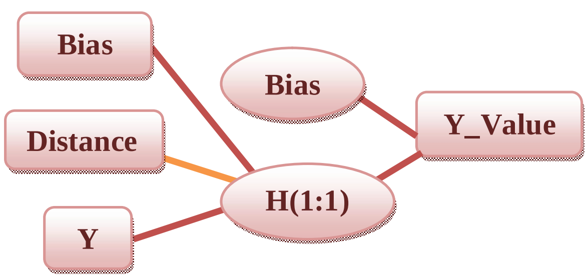

Figure 4: Multilayer Perceptron model for Angleαvs.Y Coordinate

Table 7: Model Summary

Training	Sum of Squares Error	22.622
	Relative Error	0.532
	Stopping Rule Used	One consecutive step(s) with no decrease in error (error computations are based on the testing sample)
Testing	Sum of Squares Error	10.772
	Relative Error	0.571

Similar to the case of that of the X coordinates, even the Y coordinates follows the same model.

Table 8: Parameter Estimates

Predictor		**Predicted**	
		Hidden Layer 1	**Output Layer**
		H(1:1)	Y Coordinates
Input Layer	(Bias)	0.534	
	Distance	-0.019	
	Y	0.896	
Hidden Layer 1	(Bias)		-0.272
	H(1:1)		1.128

Table 9: Independent Variable Importance

	Importance	Normalized Importance
Distance	.019	2.0%
Y	.981	100.0%

Summation of importance is 1. Hence the importance of each of the independent variables in predicting the dependent can be understood.

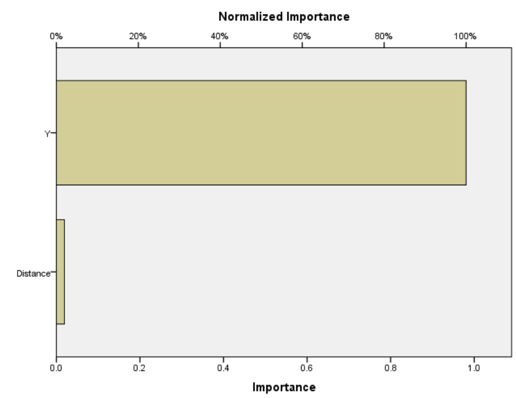

Figure 5: Variable Importance

From the figure we can clearly understand the importance of the Y coordinates of the Vanishing point for various camera angles α, and has a 100% importance level, while the distance between the sticks do not account much for the model. This agrees with the first regression equation initially derived.

Phase - II

For the same camera angle α, the distance between the sticks, and the brightness intensity was varied at small intervals to study the effect of the intensity of brightness on the detection of vanishing points in images. The results follow.

Multilayer Perceptron: Brightness – X

User and system missing values are treated as missing. Statistics are based on cases with valid data for all variables used by the procedure.

ISSN 2278-3091

International Journal of Advanced Trends in Computer Science and Engineering, Vol.2 , No.1, Pages : 171-177 (2013)
Special Issue of ICACSE 2013 - Held on 7-8 January, 2013 in Lords Institute of Engineering and Technology, Hyderabad

Table 10: Case Processing Summary

		N	Percent
Sample	Training	142	67.6%
	Testing	68	32.3%
Valid		210	100.0%
Excluded		0	
Total		210	

Table 11: Network Information

Input Layer	Covariates	1	Distance
		2	X
	Number of Units[a]		2
	Rescaling Method for Covariates		Standardized
Hidden Layer(s)	Number of Hidden Layers		1
	Number of Units in Hidden Layer 1[a]		3
	Activation Function		Hyperbolic tangent
Output Layer	Dependent Variables	1	X_Value
	Number of Units		1
	Rescaling Method for Scale Dependents		Standardized
	Activation Function		Identity
	Error Function		Sum of Squares

a. Excluding the bias unit

Synaptic Weight < 0
Synaptic Weight > 0

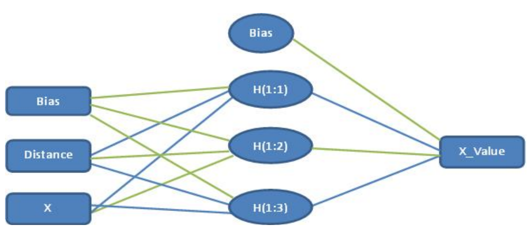

Figure 6: Hidden layer activation function: Hyperbolic tangent, Output layer activation function: Identity

Table 12: Model Summary

Training	Sum of Squares Error	31.552
	Relative Error	.725
	Stopping Rule Used	1 consecutive step(s) with no decrease in error[a]
	Training Time	0:00:00.03
Testing	Sum of Squares Error	21.762
	Relative Error	.743

Dependent Variable: X Value
a. Error computations are based on the testing sample.

The error, both sum of squares and relative is high, implying that higher and lower brightness peaks is not advisable for the detection of vanishing points.

Table 13: Parameter Estimates

Predictor		Predicted			
		Hidden Layer 1			Output Layer
		H(1:1)	H(1:2)	H(1:3)	X_Value
Input Layer	(Bias)	.086	.280	.007	
	Distance	-.184	.243	-.094	
	X	-.244	.593	-.446	
Hidden Layer 1	(Bias)				.083
	H(1:1)				-.418
	H(1:2)				.470
	H(1:3)				-.337

Table 14: Independent Variable Importance

	Importance	Normalized Importance
Distance	.289	40.7%
X	.711	100.0%

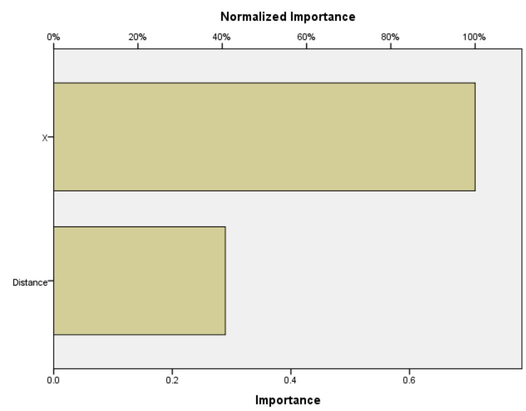

Figure 7: Normalized Importance

From the figure we can clearly understand the importance of the X(Brightness intensity in this case) of the image under study, and has a 100% importance level, while the distance between the sticks do account much for the model, partly due to the relative error.

Multilayer Perceptron: Brightness – Y
User and system missing values are treated as missing. Statistics are based on cases with valid data for all variables used by the procedure.

ISSN 2278-3091

International Journal of Advanced Trends in Computer Science and Engineering, Vol.2 , No.1, Pages : 171-177 (2013)
Special Issue of ICACSE 2013 - Held on 7-8 January, 2013 in Lords Institute of Engineering and Technology, Hyderabad

Table 15: Case Processing Summary

		N	Percent
Sample	Training	151	71.9%
	Testing	59	28%
Valid		210	100.0%
Excluded		0	
Total		210	

Table 16: Network Information

Input Layer	Covariates	1	Distance
		2	Y
	Number of Units[a]		2
	Rescaling Method for Covariates		Standardized
Hidden Layer(s)	Number of Hidden Layers		1
	Number of Units in Hidden Layer 1[a]		4
	Activation Function		Hyperbolic tangent
Output Layer	Dependent Variables	1	Y_Value
	Number of Units		1
	Rescaling Method for Scale Dependents		Standardized
	Activation Function		Identity
	Error Function		Sum of Squares

a. Excluding the bias unit

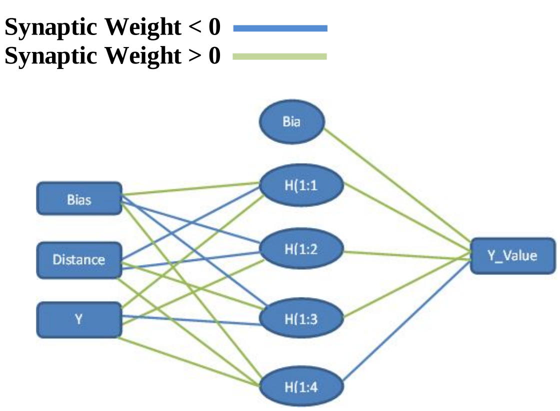

Figure 8: Hidden layer activation function: Hyperbolic tangent, Output layer activation function: Identity

Table 17: Model Summary

Training	Sum of Squares Error	27.035
	Relative Error	.581
	Stopping Rule Used	1 consecutive step(s) with no decrease in error[a]
	Training Time	0:00:00.05
Testing	Sum of Squares Error	9.178
	Relative Error	.396

Dependent Variable: Y_Value
a. Error computations are based on the testing sample.

The error, both sum of squares and relative is high, implying that higher and lower brightness peaks is not advisable for the detection of vanishing points.

Table 18: Parameter Estimates

Predictor		Predicted				
		Hidden Layer 1				Output Layer
		H(1:1)	H(1:2)	H(1:3)	H(1:4)	Y_Value
Input Layer	(Bias)	1.311	-.904	-.461	.223	
	Distance	-.127	-.064	.642	.473	
	Y	1.983	.675	-.767	.284	
Hidden Layer 1	(Bias)					.470
	H(1:1)					.894
	H(1:2)					1.018
	H(1:3)					.596
	H(1:4)					-.342

Table 19: Independent Variable Importance

	Importance	Normalized Importance
Distance	.119	13.6%
Y	.881	100.0%

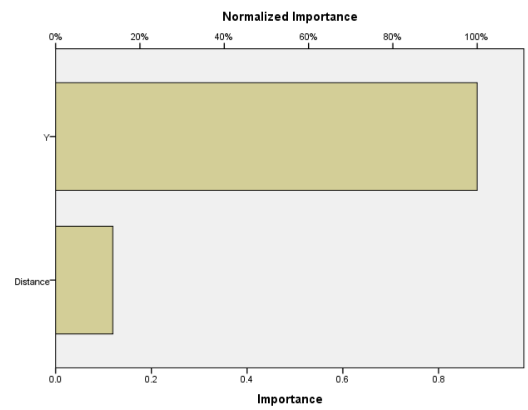

Figure 9: Normalized Importance

From the figure we can clearly understand the importance of the Y(Brightness intensity in this case) of the image under study, and has a 100% importance level, while the distance between the sticks do account some for the model, partly due to the relative error.

Conclusion

From the two models, the regression model and the neural network model, we can easily find a relationship between the camera angle and the vanishing point. Using this information and the given vanishing point various applications can be handled likeroad detection in unmanned cars, object detection in an image, as a tool to assist the visually challenged, and training canines are a few to name. Brightness plays an integral part in the detection of vanishing points, too low and very high bright photos are not advisable for processing directly.

ISSN 2278-3091

International Journal of Advanced Trends in Computer Science and Engineering, Vol.2 , No.1, Pages : 171-177 (2013)
Special Issue of ICACSE 2013 - Held on 7-8 January, 2013 in Lords Institute of Engineering and Technology, Hyderabad

Pre-processing must be done to improve the edge and horizon detection, which is yet another emerging field in research.

References

[1] Danko Antolovic, Alex Leykin, Steven D. Johnson, "*Vanishing Point: a Visual Road-Detection Program for a DARPA Grand Challenge Vehicle"*

[2] T. Lee. "*Image representation using 2D Gabor wavelets. IEEE Trans. Pattern Analysis and Machine Intelligence"*, 18(10):959–971, 1996.

[3] M. Sonka, V. Hlavac, R. Boyle, *Image Processing, Analysis and Machine Vision,* (Chapman & Hall,1993)

[4] M. Nazzal, Ibrahim M. El-Emary and Salam A. Najim, "*Multilayer Perceptron Neural Network (MLPs) For Analyzing the Properties of Jordan Oil Shale Jamal".*

[5] Wolfgang Forstner. "*Optimal Vanishing Point Detection and Rotation Estimation of Single Images from a LegolandScen"*

[6] D. G. Aguilera, J. Gomez Lahoz and J. Finat Codes "*A New Method for Vanishing Points Detection in 3D Reconstruction from A Single View*"