

## **Near Helipad, Automatic Guidance System (NeHAGS)**

**Salman Afghani, Arslan Arshad and Muhammad Ishfaq Javed**

*R&D of ES Department, APCOMS, Rawalpindi, Pakistan*

---

### **ABSTRACT**

*During the Landing Phase of the flight of a helicopter human error is the main cause of Fatal accidents. Factors such as Fatigue, Mood, Lack of attention and etc. can lead to loss of precious lives. Near Helipad Automatic Guidance System (NeHAGS) uses Machine Vision to Recognize and Track a specifically designed Landing Marker. And produces both audio and video commands for the guidance of a pilot in landing of a helicopter. Such a system not just increases safety level, it also reduces mental stress by reducing the demand of skill involved in landing a helicopter. The proposed system is very simple fast and compact because instead of relying on complicated and computationally demanding techniques such as Freak, Orb, Surf and etc. NeHAGS is based on our indigenous approach which draws its strength from the design of the Landing Marker.*

**Keywords:** Helicopter, Human error, Automatic landing, Landing marker, Landing sequence, Landing algorithm

---

### **INTRODUCTION**

The capability of Landing and takeoff without a runway, on helipads not greater than 3.0 feet in radius, on roof tops of high rise buildings and on off shore Rigs is inherent in helicopter design.

A marvelous air craft in a class of its own. But in today's world reliability and safety are the Defining Factor of performance of any machine [1]. Therefore, major emphasis of aircraft and automobile manufacturers is on developing smart device and system which will become an integral part of the product. Rather than a high end accessory which can be added on at a later stage by end users.

The current research work in the area is about using homo/heterogeneous sensors, especially cameras, both ground base and air born. In almost all the case the authors are relying on advance and complex approaches, architectures and algorithms as shown in Figure 1.



**Figure 1** : Existing system sensor-based approach

### **NeHAGS**

Our proposed System (NeHAGS) is based on Machine Vision for estimation of Range, Azimuth

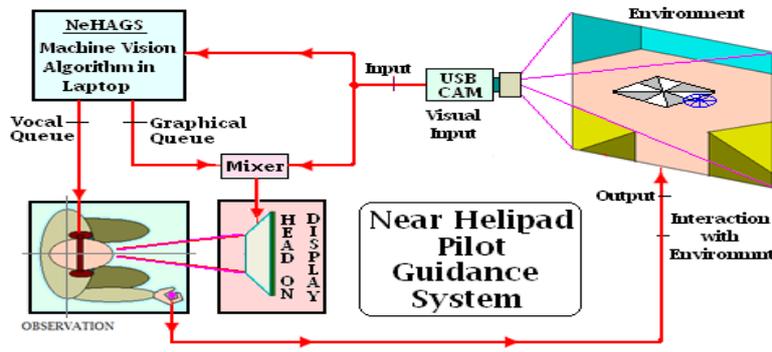


Figure 2 : Architecture diagram

and Yaw of a helicopter relative to a Landing Marker using Monocular Camera. A simple yet reliable strategy is used for keeping the Guidance System Invariant to Rotation, Scale and Affine transformation. And this provided the opportunity to bypass complex and lengthy algorithms such as Sift, Orbit and Surf as shown in Figure 2.

**Landing marker and elliptical kernel design for landing marker detection**

The marker is based on black and white, eight-sided British flag sort of Rectangular surface lying on a horizontal plane.

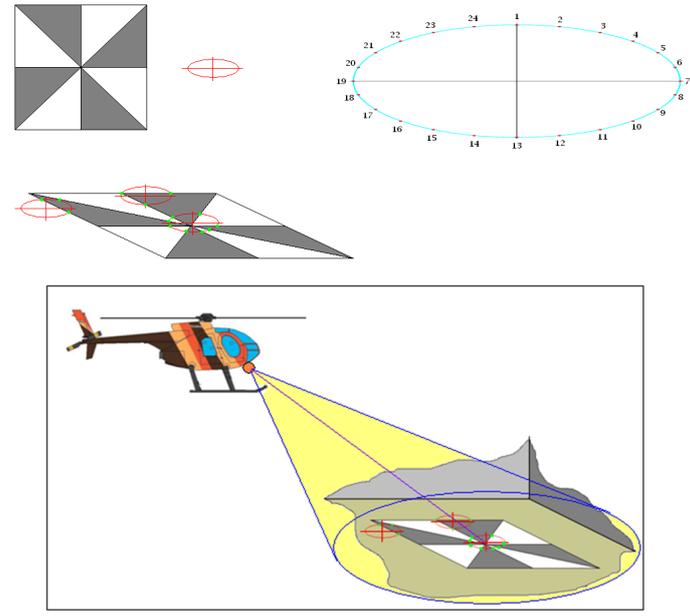


Figure 3: Marker design and transition points

NeHAGS continuously seeks for the Landing Marker (LaMa) using its Elliptical Marker Detector Kernel (EMDK). The moment the centers of the two coincide with an acceptable degree of error of not more than 20%, the EMDK will encounter 8 transitions of intensity along its parameter as shown in Figures 3 and 4a.

But in realistic conditions, noise cannot be underestimated. Therefore, flexibility was incorporated into the Marker Detection Criteria for robust performance. Last but not the least of problems encountered during development of EMDK design, the problem was the Parallel Lines False Detection Dilemma as shown in Figure 5. The dilemma was taken care of by using 2 concentric EMDK's instead of one's shown in Figure 4b.

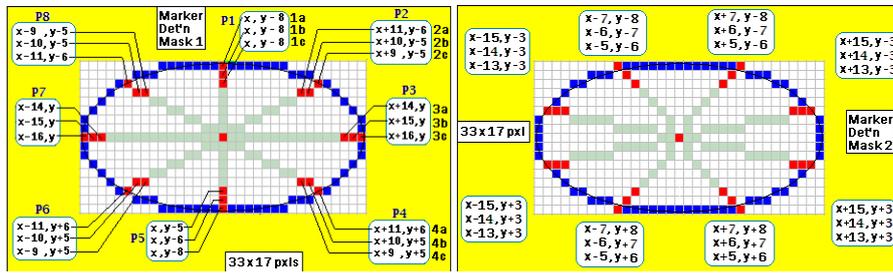


Figure 4a: EMDK 1 Figure 4b. EMDK 2

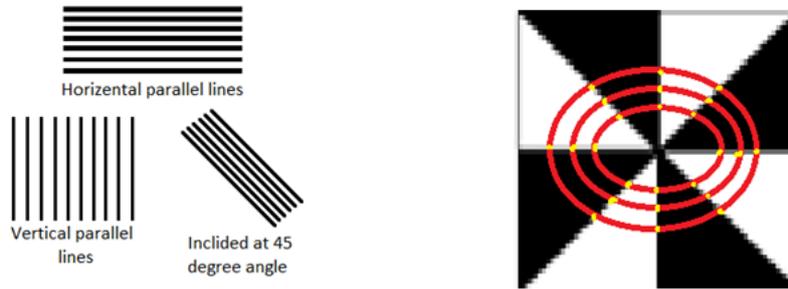


Figure 5. Parallel lines issue

Baring in mind all the above considerations, a list of conditions for robust detection of Landing Marker (LaMa) is presented below:

1. Not more than 10 & not less than 6 transition pixels should be encountered along the parameter of the primary ellipse.
2. At least 1 & not more than 2 transitions should be present in each of the 4 quadrants of the primary ellipse.
3. At least 4 transition locations on the perimeter of main ellipse should coincide with similar 4 locations on the perimeter of the secondary ellipse.

**Landing sequence**

As soon as the Landing Marker is insight of NeHAGS (~ 100 m radius), the System invites the pilot to take advantage of Machine Vision based Landing Assistance System. The pilot can now activate and use NeHAGS as a Cyber Landing Assistant which provides Visual as well as Audio Cues to Guide the pilot towards the landing pad. The sequence in which NeHAGS functions, is listed below:

**Frame capture (FrCa)**

From the RGB buffer available to NeHAGS, only Red Channel (640 × 460 pixels) is captured and stored into a 2D Grey Level Array as shown in Figures 6a and 6b.

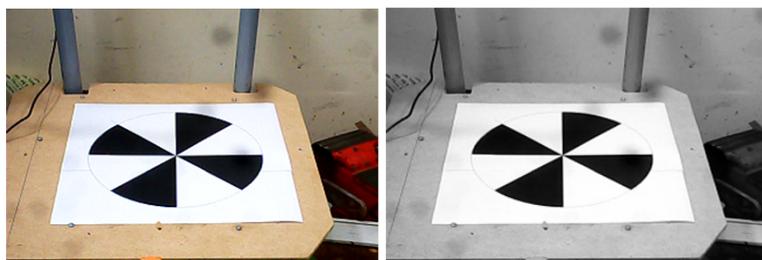
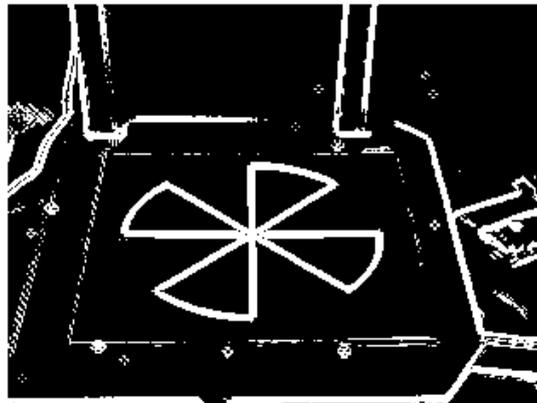


Figure 6a: Original image Figure 6b: Gray scale image

Each of captured gray frame is subjected to our slop strength edge detector (SSED) as shown in Figure 7. The SSED filter creates and indexed array, containing location of only those pixels which belongs to certain edges in the captured frame [2,3].



**Figure 7:** After SSED filter high contrast zone locator (hczl)

A 64\*64 pixel, half stepping (32 pixel/ step) rectangular locator scanned the index array and quantifies the number and edge pixels in each rectangular zone position. Three zones with highest edge pixel or identified a location of their center are stored in 6 zones register (Rz1x, Rz1y), (Rz2x, Rz2y) and (Rz3x, Rz3y).

#### **Marker detection elliptical kernel (MaDEK)**

For accuracy these three zones are broken down further in to 3\*3 sub zones. And the centers of these sub-zones are superimposed on the original grey frame as centers of the Marker Detection Elliptical Kernel (MaDEK).

#### **Max. probability spatial filter (FSMP)**

Out of 3\*3\*3 MaDEK center only those pixels are filtered out, which satisfy the MaDEK selection criteria. The locations of only the selected pixels are stored in register (Rsf1x, Rsf1y), (Rsf2x, Rsf2y), (Rsf3x, Rsf3y). These landing marker candidates are named as Primary Landing Marker Candidates (LMCP).

#### **Max. probability temporal filter (FTMP)**

The selected Primary Landing Marker Candidates (LMCP) are further scrutinized by FSMT, till only one pixel is left to represent the true Landing Marker. This task is accomplished by repeated execution of steps 1 till 5, eight times. The coordinates of 'n' number of Secondary Landing Marker Candidates (LMCS) pixels acquired after 8 consecutive executions of steps 1 till 5, are tested for Proximity to the coordinates of the 3 Primary Landing Marker Candidates (LMCP) pixels. For this very purpose, three Neighborhood Strength Counters (CNS 1, CNS 2, CNS 3) are maintained (one for each LMCP). The LMCP having the closest of neighbors among the LMCS pixels, is declared, the True Landing Marker (LMT).

#### **Off-center distance calculator (OCDC)**

Cartesian Distance between the Optical Axis of the camera and the coordinates of (LMT) is calculated in terms of x & y components of the Offset.

#### **Azimuth calculator**

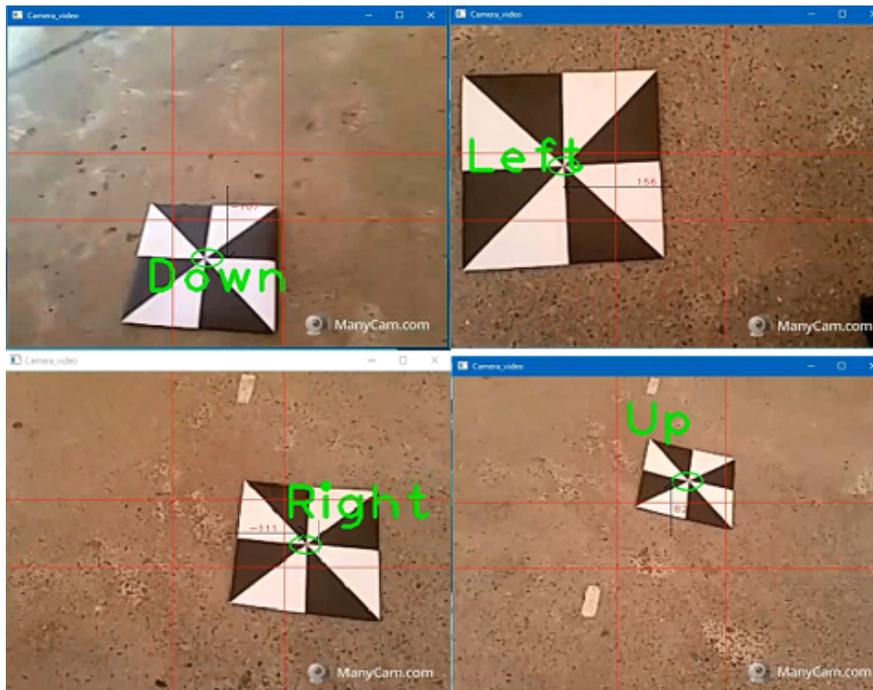
The Cartesian Distance is converted into Polar Coordinates to determine the Angular Offset (Azimuth).

#### **Range calculator**

Pythagoras is used to convert x and y components of distance into Hypotenuse (Range).

**Altitude estimator**

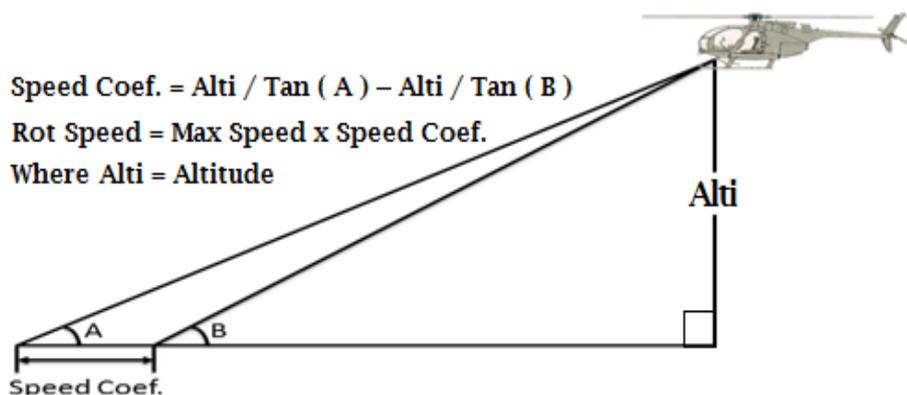
A set of 8 concentric Elliptical Marker Detecting Kernels (EMDK) are used to measure the size of the True Landing Marker (LMT). And size corresponds to the Distance of LMT from the observer (on board camera). The moment the size of LMT falls with-in the smallest of the 8 EMDK's, NeHAGS goes for Blind Landing to point 11b. But if Helicopter is still at a distance greater than least, then NeHAGS goes for another round of Altitude, Range and Azimuth correction to point 11a.



**Figure 8:** NeHAGS tracking the LaMa and generating simplistic instructions for the pilot

This point has two parts s follows

- a. **Azimuth stepper:** The view port of  $640 \times 460$  pixels is segmented into Graded Rectangular Regions. Each Rectangle represents a zone of a particular distance of a portion of Ground from the observer (on-board camera). And subsequently each Zone is assigned an Azimuth Centering Speed ('nn' pixels/s). This evaluation is based on the angular orientation and Altitude of camera in reference to the Ground Surface as shown in Figures 8 and 9. The angle of optical axis of camera in reference to Ground remains Constant, whereas Altitude varies and so does the actual size of the area of horizontal plane (ground) under observation in the view port.



**Figure 9:** Altitude and angular orientation of camera

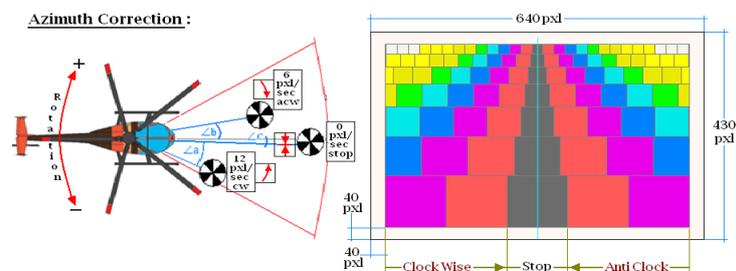
The higher the Altitude, the further and bigger the actual horizontal surface being observed. And subsequently, slower the speed of tracking. The relationship between various Altitudes and desired Marker Tracking speed is Table 1.

**Table 1.** Altitudes and desired marker tracking speed measurements

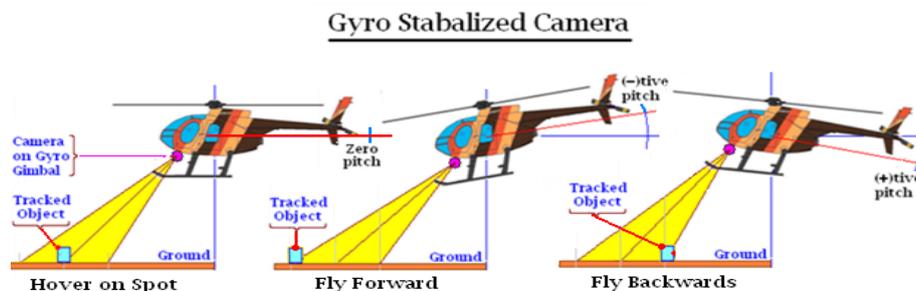
S/N	Angle		Altit.=55		Altit.=45		Altit.=35		Altit.=25		Altit.=15		Altit.=05	
	A	B	Speed Coef	Rot Speed										
1	45	40	10.546	4.266	8.628	5.215	6.711	6.705	4.793	9.387	2.876	15.645	0.958	46.935
2	40	35	13.001	3.461	10.637	4.230	8.273	5.438	5.909	7.614	3.545	12.690	1.181	38.071
3	35	30	16.71	2.692	13.675	3.290	10.636	4.230	7.597	5.922	4.558	9.8915	1.519	29.614
4	30	25	22.685	1.983	18.560	2.424	14.435	3.117	10.311	4.364	6.186	7.2735	2.062	21.820
5	25	20	33.163	1.356	27.133	1.658	21.103	2.132	15.074	2.985	9.044	4.9753	3.014	14.926
6	20	15	54.151	0.831	44.305	1.015	34.460	1.305	24.614	1.828	14.768	3.0470	4.922	9.141
7	15	10	106.65	0.421	87.265	0.515	67.873	0.663	48.480	0.928	29.088	1.5470	9.696	4.641

The Graded Azimuth Correction Frame is partitioned in the middle into 2 Horizontally adjacent sub-frames. If Marker is detected anywhere to the left of vertical partitioning line, then azimuth correction is performed by rotating the Helicopter clock-wise at speeds graded in the Rectangular Zones. And visa-versa, if landing marker is located to the right of the vertical partitioning [4].

The Grey zone along the length of the vertical partitioning line, is the Zero Velocity zone. The helicopter stops correcting Azimuth, once ‘LaMa’ enters this zone. It is obvious that the upper portion of the View port which represents the Farthest section of Ground Surface, is Graded with Lower Velocities, whereas the lower Portion which is closer the observer is graded with Higher Velocities. The optimization in proportionality among the rectangular regions were achieved experimentally. The above described Marker Tracking approach was adopted for reduction in algorithm complexity and computational intensity.



**Figure 10:** Azimuth correction



**Figure 11:** Gyro stabilized camera

Last but not the least to be mentioned is the fact, that a pitch stabilizing, motorized, miniature gyro-camera gimble as shown in Figures 10 and 11 had to be added to the on-board avionics of the helicopter for increasing the reliability of the system to almost 80%.

### Blind landing stepper (BLaS)

Blind Landing Stepper is totally independent of Machine Vision section of NeHAGS. It relies on a pair of electronic sensors. One of them is an IR-Proximity sensor and the other, a Tactile touch-down sensor. The IR Proximity sensor is in place to break the rapid decent and avoid heavy impact upon landing. During the decent, when this sensor gets tripped, the system instantaneously boosts the speed of the main lift rotor by 25%, which produces a strong burst of air in the form of an air-cushion for the helicopter. And this extra boost keeps on increasing till the helicopter climbs to a height where the proximity sensor is turned-off. And when that happens, the system starts gradually stepping down the speed of the main lift rotor,

till the touch-down Tactile sensors make contact with the ground. Upon touch-down, the power to the main rotors is cut-off.

### Altitude/range flip-flop

Here a Flip-Flop switches turns between execution of Range stepper or altitude stepper algorithm. If logic state of Flip-Flop is '0', then Altitude stepper at point 13 is selected. And if '1' then, Range Stepper at point 14.

### Range stepper

Similar to Azimuth Stepper, Range Stepper has the view port segmented into Graded Rectangular Regions 'GRR'. But Range Stepper swings into action only, if the Marker falls within the perimeters of the middle most GRR's lying over the vertical line dividing the view port into two equal halves. Drifting of LaMa into any GRR other than the ones lying over the mid-vertical line, will force NeHAGS, to branch off to the Azimuth Stepper Algorithm. Range correction is performed by pitching down the main lift rotor of helicopter to fly in the forward direction at speeds dictated by the GRR, coinciding with the LaMa as shown in Figure 12. The progress in the forward direction is halted by neutralizing the pitch as soon as LaMa enters the Stop GRR. Upon reaching the Stop GRR, NeHAGS branches off to Altitude Stepper Algorithm. But sometimes, NeHAGS overshoots the Stop GRR, and LaMa slips down into the Reverse GRR (largest & lowest). And when that happens, the helicopter Pitches up the main lift rotor to fly in the backwards direction till the Stop GRR is reached.

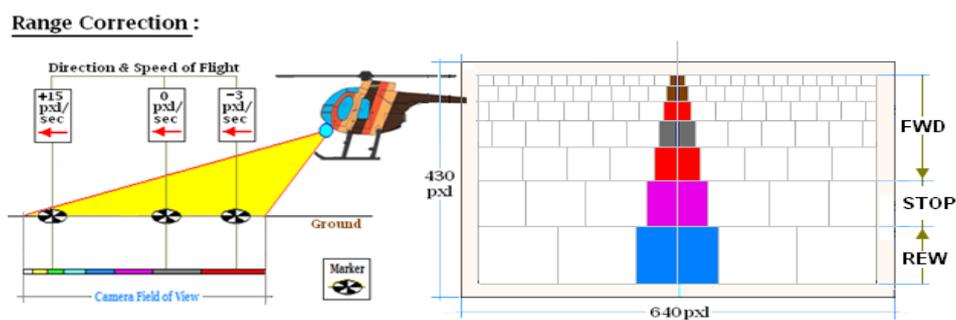


Figure 12: Range correction

### Altitude stepper

Similar to Range Stepper, Altitude Stepper has the view port segmented into Graded Rectangular Regions 'GRR'. And similarly, Altitude Stepper swings into action only, if LaMa falls within the perimeters of the middle most GRR's lying over the vertical line dividing the view port into two equal halves as shown in Figure 13. Drifting of LaMa into any GRR other than the ones lying over the mid-vertical line, will force NeHAGS, to branch off to the Azimuth Stepper Algorithm. Altitude correction is performed by Reducing Thrust of Main lift rotor of helicopter to Decent at speeds dictated by the GRR, coinciding with the LaMa. The Decent is halted as soon as LaMa climbs up to the Stop GRR. Upon reaching the Stop GRR, NeHAGS branches off to Altitude Stepper Algorithm. But sometimes, NeHAGS overshoots the Stop GRR, and LaMa climbs over into the Acent GRR (smallest & highest). And when that happens, the helicopter increases Thrust in Main lift rotor to gain Altitude till the Stop GRR is reached [5-7].

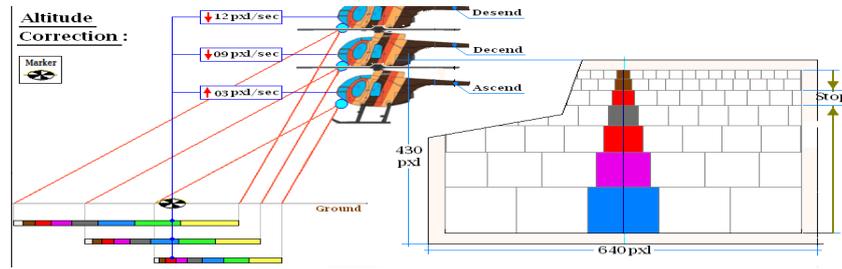


Figure 13: Altitude correction

Eventually a robust working, Automatic Helicopter Pilot Guidance System was successfully implemented. Proof of concept was demonstrated using a USB camera held in the hands of a Blind Folded operator along with a Minimalistic NEHAGS Algorithm running in a Laptop.

The Minimalistic NEHAGS algorithm is a minimum component derivative of the main NEHAGS algorithm elaborated previously. The Minimalistic Version can be broken down to three functional modules:

**Main caller**

Depending upon the success in detection of the Landing Marker, the Main Caller switches between Full (640 × 480) Frame and Small (100 × 100 pixel) Frame Landing Marker tracking.

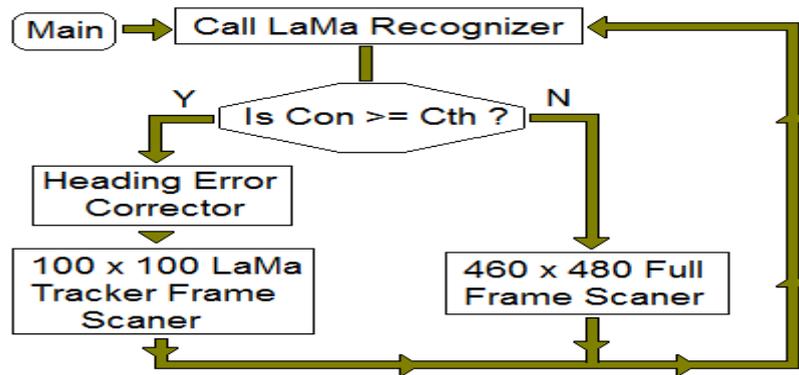


Figure 14: Main calling algorithm

- a. **Landing marker recognizer:** This Algorithm superimposes 2 LaMa Detector Masks LDM1 and LDM2 as shown in Figures 14 and 15. One after another in an attempt to detect the existence of a LaMa at the current location on a captured frame.

The Design of LaMa detector has been elaborated earlier in this paper, but exact technique is presented below. Two Coordinate tables TBL1[i] and TBL2[i] are used to maintain the coordinates of 3 Concentric Ellipses per table, where each ellipse is represented by 8 evenly placed points along the perimeter of each ellipse.

Intensity of pixels at coordinates maintained in table TBL1[i] are read into Intensity Array Ai1[i] and likewise the Intensities of pixels at locations in table TBL 2 are read into Array Ai2[i].

Intensity of each point stored in Array Ai1[i] and Ai2[i] is compared to its immediate neighbor along the perimeter of each of the 3 ellipses in a clockwise manner.

All these contrast values are stored into 2 contrast Arrays Ac1[i] and Ac2[i], respectively.

A simple Dynamic Thresh Hold value  $THC = \frac{3}{4} \text{ Max Contrast}$  is derived from entries in the 2 Contrast Arrays, Ac1[i] and Ac2[i]. But THC has to be  $\geq 15$  (Noise Level). If not, the pass flag is cleared (Fpas=0) before Return from call.

All the entries in contrast Array Ac1[i] are measured up to THC. If entry in Ac1[i] is  $\geq$  THC, then a counter called Valid Point is incremented (Cvp+1) otherwise next entry is tested against THC. This process continues till the last entry in Ac1[i] is through. And when that happens, the Valid Point Counter Cvp is tested. If Cvp=18, then it is presumed that considerable number of transitions are detected to support the presence of a LaMa at the current location. Thus, a Pass Flag is set (Fpas=1) before Return. If not then same procedure of testing entries against THC is followed for Contrast Array Ac2 [i].

But if this time the procedure dose not yield Positive result then of course LaMa is not present and the Pass Flag is cleared Fpas=0 before Return and Vice Versa.

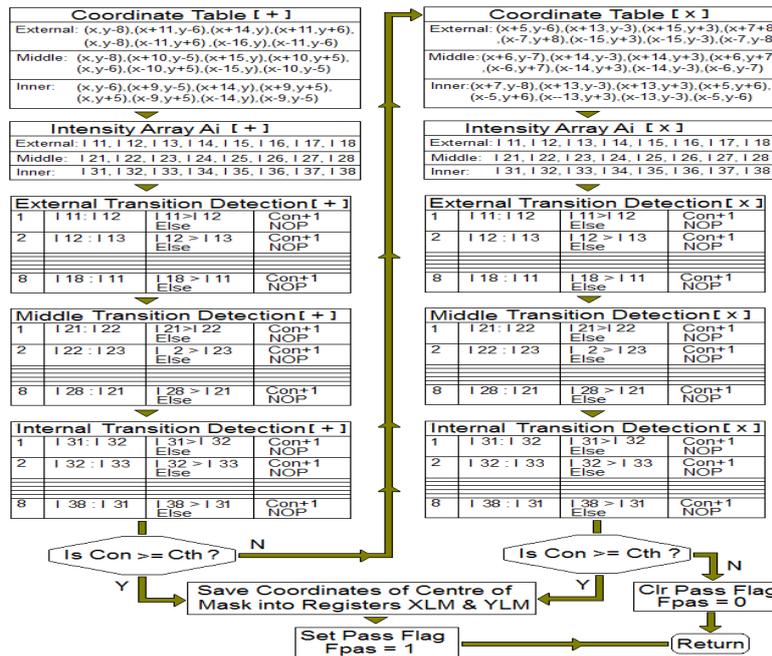


Figure 15: Landing marker recognizer

**Nugatory distance calculator (NDC)**

Depending upon the Distance between the Optical Axis of the Camera and the center of the Landing Marker, the NDC module provides Audio/Visual Guidance to the Pilot, for a 5 speed (Slow/Slow-Mod/Moderate/Mod-Fast/Fast) correction of Azimuth and Pitch of the helicopter in an attempt to get the Heading of the Helicopter aligned with the center of the Landing Marker. This module will use Head Error Corrector algorithm as shown in Figure 16.

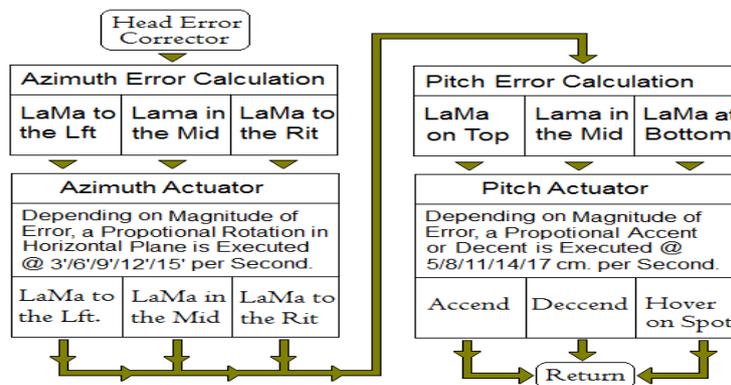


Figure 16: Head error corrector algorithm

### CONCLUSION

Eventually a robust working, Automatic Helicopter Pilot Guidance System was successfully implemented. Proof of concept was demonstrated using a USB camera held in the hands of a Blind Folded operator along with a Minimalistic NEHAGS Algorithm running in a core 5 Laptop.

### REFERENCES

- [1] Patruno, C., et al., *Int J Adv Robot Syst*, **2017**. 14(5): p. 1-15.
- [2] Masuko, K., et al., *IEEE*, Sendai, Japan, **2010**.
- [3] Cicirelli, G., D’orazio, T. and Distanto, A., *Exp Theor Artif Int*, **2003**.
- [4] Patruno, C., Marani, R., Nitti, M., *IEEE, Trans Int Trans Syst*, **2015**.
- [5] Chialastri, A., InTechOpen, **2012**.
- [6] Zhang, Z., *IEEE*, Kerkyra, Greece, **1999**. p. 666-673.
- [7] Lange, S., Sünderhauf, N. and Protzel, P., In SIMPAR, Venice, Italy, **2008**, p. 482-491.