

DESAIN KONTROL *PATH FOLLOWING* QUADCOPTER DENGAN ALGORITMA *LINE OF SIGHT*

Anggara Trisna Nugraha ¹⁾

¹⁾Jurusan Teknik Elektro, Fakultas Teknik, Institut Teknologi Sepuluh Nopember
Jl. Raya ITS, Keputih, Sukolilo, Keputih, Sukolilo, Surabaya, Jawa Timur 6011
Email : anggaratnugraha@yahoo.co.id

Abstrak . Perkembangan *Quadcopter* sudah semakin luas fungsinya, bahkan untuk tugas-tugas militer di udara yang berbahaya, misalnya operasi pencarian, pengintaian, penyelamatan dan juga bisa diaplikasikan untuk tugas ilmiah seperti pemetaan kondisi kecepatan angin, mendeteksi sumber radiasi, pengawasan, pemeliharaan dan survei. *Quadcopter* merupakan sistem nonlinear dengan multiple input multiple output dan sistem dengan persoalan kestabilan, sehingga rentan terhadap gangguan eksternal. Karakteristik ini menyebabkan adanya kesulitan dalam pengendalian gerak tracking secara otomatis dan mengatur arah hadap. Berdasarkan permasalahan tersebut, pada penelitian ini disajikan desain kontrol tracking pada bidang horizontal dengan menambahkan algoritma *Line of Sight*, agar dalam melakukan *path following* konvergen menuju nol dan dapat mengatasi gangguan arus laut yang merubah parameter dari *quadcopter* dalam melakukan gerak pada bidang horizontal. Hasil simulasi menunjukkan bahwa metode kontrol yang digunakan mampu membawa sudut yaw, pitch dan roll pada nilai yang diharapkan. *Quadcopter* dapat melakukan tracking secara otomatis pada kondisi tanpa gangguan dengan rata-rata *cross track error* sebesar 0.8 meter.

Kata kunci: *Quadcopter*, Tracking Control, Line Of Sight, Path Following.

1. Pendahuluan

Saat ini teknologi *Unmanned Aerial Vehicle* (UAV) telah banyak digunakan dalam keperluan pengawasan, pencarian dan penyelamatan. Seiring berkembangnya metode yang digunakan untuk menangani permasalahan kontrol pada *quadcopter* yang mana ^[1], teknik kontrol *nonlinear* yang digunakan untuk kontrol *hover*. Metode *nonlinear backstepping* lebih bisa mendekati karakteristik sistem *nonlinear*. Sehingga, langkah perhitungan yang digunakan lebih rumit. Walaupun dengan perhitungan rumit tersebut *quadcopter* mampu melakukan *path following* sesuai refrensi dari kondisi nilai x , y dan z . Namun pada saat *quadcopter* diberi gangguan berupa angin yang konstan maka respon yang dihasilkan mengalami *ripple* dan tidak *robust*. Pada ^[2] disajikan algoritma *Line Of Sight* (LOS) yang memudahkan untuk mengatur arah hadap suatu *plant* agar konvergen menuju *path* yang sudah didefinisikan, yaitu dengan membuat *cross-track error* yang dituju.

Berdasarkan pada permasalahan dan solusi diatas, maka ide dalam penelitian yang akan diajukan adalah merancang desain metode kontrol optimal untuk mengontrol *quadcopter* agar dapat melakukan *path following* dengan jalur yang sudah ditentukan dan menambahkan algoritma *Line Of Sight* (LOS) yang digunakan untuk dapat menghasilkan kesesuaian terhadap suatu bentuk *path* sebagai sinyal referensi dengan mengatur arah hadap serta *path error* sekecil mungkin.

2. Pembahasan

2.1. Model Dinamika Translasi *Quadcopter*

Suatu gaya yang dihasilkan oleh keempat motor penggerak yang dipasang baling-baling pada *quadcopter*. Persamaan berikut: menunjukkan gaya-gaya yang terjadi :

$$U_1 = F_1 + F_2 + F_3 + F_4 \quad (1)$$

$$U_2 = F_2 - F_4 \quad (2)$$

$$U_3 = F_1 - F_3 \quad (3)$$

$$U_4 = F_1 + F_3 - F_2 - F_4 \quad (4)$$

Model Dinamika Rotasi Quadcopter

Persamaan dinamika dari quadcopter pada gerak translasi dan rotasi adalah

$$\ddot{x} = (\cos \phi \sin \theta \cos \psi + \sin \phi \sin \psi) \frac{U_1}{m} \quad (5)$$

$$\ddot{y} = (\cos \phi \sin \theta \cos \psi - \sin \phi \sin \psi) \frac{U_1}{m} \quad (6)$$

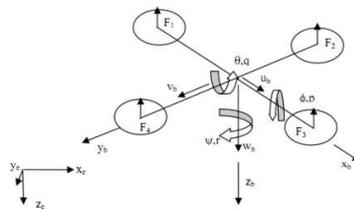
$$\ddot{z} = -g + (\cos \phi \cos \theta) \frac{U_1}{m} \quad (7)$$

$$\ddot{\phi} = \frac{J_{yy} - J_{zz}}{J_{xx}} qr + \frac{J_r}{J_{xx}} q\Omega + \frac{U_2}{J_{xx}} \quad (8)$$

$$\ddot{\theta} = \frac{J_{zz} - J_{xx}}{J_{yy}} pr - \frac{J_r}{J_{yy}} p\Omega + \frac{U_3}{J_{yy}} \quad (9)$$

$$\ddot{\psi} = \frac{J_{xx} - J_{yy}}{J_{zz}} pq + \frac{U_4}{J_{zz}} \quad (10)$$

dengan m adalah massa, g adalah gravitasi, l adalah jarak antara pusat massa quadcopter dengan sumbu rotasi *propeller*, d adalah koefisien gaya drag, dan I_x, I_y, I_z adalah inersia pada sumbu x, y, z berturut-turut.



Gambar 1. Gambar Semua *State* dari Quadcopter (e untuk *Frame* Bumi dan b untuk *Frame* Badan Quadcopter) [3]

2.2. Linearisasi Model Dinamika Quadcopter

Pada bagian ini akan dijelaskan mengenai linearisasi dinamika quadcopter, dimana konsep dinamika sudah diterangkan pada sub-bab dinamika quadcopter. Nilai parameter, yang ditulis dalam tabel 1. variable *state* v , yang akan digunakan sebagai representasi dinamika pada aktuator quadcopter seperti berikut:

$$v = -\frac{\omega}{s + \omega} u \quad (11)$$

Tabel 1. Parameter Sistem Quanser Qball-X4³⁾

No	Parameter	Simbol	Nilai
1	Massa	m	3,499 kg
2	Gravitasi	g	9,81 kg/m ²
3	Moment Inersia pada sumbu X	J_{xx}	0.03 kg.m ²
4	Moment Inersia pada sumbu Y	J_{yy}	0.03 kg.m ²
5	Moment Inersia pada sumbu Z	J_{zz}	0.04 kg.m ²
6	Jarak rotor dari pusat massa	l	0.2 m
7	Gaya drag	d	$3,13 \times 10^{-5}$
8	Gaya trust	b	$7,5 \times 10^{-7}$
9	<i>Bandwith</i> aktuator	ω	15 rad/s
10	Konstanta gaya dorong	K	120 N

Pemodelan *Linearisasi* sudut *Roll* dan *Pitch*

Asumsikan rotasi disekitar sumbu x dan y adalah *decoupled*, pergerakan dalam sumbu *roll/pitch*. Gaya angkat yang dihasilkan oleh setiap motor dapat dikalkulasi dan menggunakan *input* masing-masing yang berhubungan. Jika konstanta drag diabaikan dan quadcopter dianggap dalam kondisi melayang dimana percepatan quadcopter mendekati

$$\phi \ll 0.1 \Rightarrow \sin(\phi) \cong 0, \cos(\phi) \cong 1 \quad (12)$$

$$\theta \ll 0.1 \Rightarrow \sin(\theta) \cong 0, \cos(\theta) \cong 1$$

Bentuk kinematika sederhana *roll*, *pitch* dan *yaw*

$$\begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} -p \\ q \\ r \end{bmatrix} \quad (13)$$

Sudut *roll/pitch* yang dimisalkan θ , dapat diformulasikan menggunakan dinamika sebagai berikut:

$$J\ddot{\theta} = \Delta FI \quad (14)$$

dengan $J = J_{roll} = J_{pitch}$, l adalah jarak propeller terhadap pusat gravitasi, dan $\Delta F = F_1 - F_2$ terdapat perbedaan antara gaya yang telah dibangkitkan oleh kedua motor. Perbedaan gaya yang dihasilkan oleh perbedaan *input* untuk kedua motor yaitu,

$$\Delta u = u_1 - u_2 \quad (15)$$

Bentuk persamaan *state space* untuk model linier dinamika *roll* dan *pitch* dapat dinyatakan:

$$\begin{bmatrix} \dot{\phi} \\ \ddot{\phi} \\ \dot{v} \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & \frac{IK_T}{J} \\ 1 & 0 & -\omega \end{bmatrix} \begin{bmatrix} \phi \\ \dot{\phi} \\ v \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ \omega \end{bmatrix} \Delta u_2 \quad (16)$$

$$\begin{bmatrix} \dot{\theta} \\ \ddot{\theta} \\ \dot{v} \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & \frac{IK_T}{J} \\ 1 & 0 & -\omega \end{bmatrix} \begin{bmatrix} \theta \\ \dot{\theta} \\ v \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ \omega \end{bmatrix} \Delta u_1 \quad (17)$$

Pemodelan *Linearisasi X* dan *Y*

Gaya angkat total dan perubahan sudut *roll/pitch* yang terjadi sepanjang sumbu X dan Y . Jika diasumsikan sudut *yaw* adalah nol maka, dinamika gerak dalam sumbu X dan Y dapat ditulis sebagai berikut:

$$\ddot{x} = \frac{4F}{m} (\sin \phi \sin \psi + \cos \phi \sin \theta \cos \psi) \quad (18)$$

$$\ddot{y} = \frac{4F}{m} (-\sin \phi \sin \psi + \cos \phi \sin \theta \cos \psi) \quad (19)$$

Percepatan sudut *roll* dan *pitch* mendekati nol yang diasumsikan, maka dengan mensubstitusikan, model posisi sumbu X dan Y dapat ditulis menjadi:

$$\ddot{x} = \frac{4F}{m} v \sin \theta \approx \frac{4K_T}{m} v \theta \quad (20)$$

$$\ddot{y} = \frac{4F}{m} v (-\sin \phi) \approx -\frac{4K_T}{m} v \phi \quad (21)$$

Model linier dinamika posisi yang didapat pada sumbu X dan Y dalam bentuk *state space* adalah sebagai berikut:

$$\begin{bmatrix} \dot{x} \\ \ddot{x} \\ \dot{v} \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & \frac{4K_T}{J} \theta \\ 0 & 0 & -\omega \end{bmatrix} \begin{bmatrix} x \\ \dot{x} \\ v \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ \omega \end{bmatrix} u \quad (22)$$

$$\begin{bmatrix} \dot{y} \\ \dot{z} \\ \dot{v} \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & \frac{4K_T}{J}\phi \\ 0 & 0 & -\omega \end{bmatrix} \begin{bmatrix} y \\ z \\ v \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ \omega \end{bmatrix} u \quad (23)$$

dimana $X = [x, y, z, u, v, w, \phi, \theta, \psi, p, q, r]^T$ (24)

$$= [x, y, z, \dot{x}, \dot{y}, \dot{z}, \dot{\phi}, \dot{\theta}, \dot{\psi}, \dot{p}, \dot{q}, \dot{r}]^T$$

dan $U = [U_1, U_2, U_3, U_4]$ (25)

2.3. Algoritma Guidance

Hukum LOS (Line of Sight) pada Persamaan Steering

Strategi berdasarkan *enclosure* digunakan untuk mengarahkan $e(t)$ menuju nol, kemudian mengarahkan vektor kecepatan menuju titik perpotongan $P_{los}^n = [x_{los}, y_{los}]^T$ agar sesuai dengan arah dari *path*, dimana *path* tersebut secara implisit didefinisikan oleh urutan di mana *waypoints* yang sudah didefinisikan. *Path* tersebut melibatkan secara langsung penetapan $\chi_d(t)$.

$$\tan(\chi_d(t)) = \frac{\Delta y(t)}{\Delta x(t)} = \frac{y_{los} - y(t)}{x_{los} - x(t)} \quad (26)$$

Koordinat pusat suatu kendaraan $\{b\}$ didefinisikan dengan $P^n = [x, y]^T$ dan digunakan sebuah lingkaran dengan radius $R > 0$ yang ditarik dari titik pusat kendaraan $\{b\}$. Lingkaran tersebut akan memotong suatu *path* pada dua titik, dimana salah satu titiknya adalah (x_{los}, y_{los}) . Perhitungan dua titik LOS $P_{los}^n = [x_{los}, y_{los}]^T$ yang tidak diketahui, dapat dicari dengan menyelesaikan dua persamaan dibawah ini:

$$\begin{aligned} [x_{los} - x(t)]^2 + [y_{los} - y(t)]^2 &= R^2 \\ \tan(\alpha_k) &= \frac{y_{k+1} - y_k}{x_{k+1} - x_k} \\ &= \frac{y_{los} - y_k}{x_{los} - x_k} = \text{konstan} \end{aligned} \quad (27)$$

dimana persamaan merupakan persamaan *pythagoras*, sedangkan persamaan menunjukkan sudut kemiringan dari *path* terhadap sumbu x bumi sebesar α_k . Kemiringan diantara dua *waypoints* tersebut adalah konstan dan berlaku juga besar kemiringannya untuk titik-titik diantaranya. Proses perhitungan *enclosure* secara analitik berdasarkan [17] dapat dijabarkan dengan:

Argumen 1: untuk $|\Delta x| > 0$, maka persamaan

$$y_{los} = \frac{\Delta y}{\Delta x} (x_{los} - x_k) + y_k \quad (28)$$

dengan $\Delta x := x_{k+1} - x_k$ dan $\Delta y := y_{k+1} - y_k$ merupakan perbedaan posisi x dan y diantara dua *waypoints*.

Koordinat pusat suatu kendaraan $\{b\}$ didefinisikan dengan $P^n = [x, y]^T$ dan digunakan sebuah lingkaran dengan radius $R > 0$ yang ditarik dari titik pusat kendaraan $\{b\}$. Lingkaran tersebut akan memotong suatu *path* pada dua titik, dimana salah satu titiknya adalah (x_{los}, y_{los}) . Perhitungan dua titik LOS $P_{los}^n = [x_{los}, y_{los}]^T$ yang tidak diketahui, dapat dicari dengan menyelesaikan dua persamaan dibawah ini:

$$[x_{los} - x(t)]^2 + [y_{los} - y(t)]^2 = R^2 \quad (29)$$

$$\begin{aligned} \tan(\alpha_k) &= \frac{y_{k+1} - y_k}{x_{k+1} - x_k} \\ &= \frac{y_{los} - y_k}{x_{los} - x_k} = \text{konstan} \end{aligned} \quad (30)$$

dimana persamaan (29) merupakan persamaan *pythagoras*, sedangkan persamaan (30) menunjukkan sudut kemiringan dari *path* terhadap sumbu x bumi sebesar α_k . Kemiringan diantara dua *waypoints* tersebut adalah konstan dan berlaku juga besar kemiringannya untuk titik-titik diantaranya. Proses perhitungan *enclosure* secara analitik dapat dijabarkan dengan:

Argumen 1: untuk $|\Delta x| > 0$, maka persamaan

$$y_{los} = \frac{\Delta y}{\Delta x} (x_{los} - x_k) + y_k \quad (31)$$

dengan $\Delta x := x_{k+1} - x_k$ dan $\Delta y := y_{k+1} - y_k$ merupakan perbedaan posisi x dan y diantara dua *waypoints*. Agar lebih mudah dalam melakukan proses perhitungan maka perlu dilakukan proses notasi baru sebagai permisalan:

$$d := \frac{\Delta y}{\Delta x}, e := x_k, f := y_k \quad (33)$$

dengan menjabarkan persamaan berikut, maka :

$$x_{los}^2 - 2xx_{los} + x^2 + y_{los}^2 - 2yy_{los} + y^2 = R^2 \quad (34)$$

dimana:

$$\begin{aligned} y_{los}^2 &= \left[\left(\frac{\Delta y}{\Delta x} \right) (x_{los} - x_k) + y_k \right]^2 \\ &= [dx_{los} + (f - de)]^2 \\ &= (dx_{los} + g)^2 \\ &= d^2 x_{los}^2 + 2dgx_{los} + g^2 \end{aligned} \quad (35)$$

dimana:

$$g := f - de = y_k - \left(\frac{\Delta y}{\Delta x} \right) x_k \quad (37)$$

kemudian pertimbangkan:

$$\begin{aligned} 2yy_{los} &= 2y(dx_{los} + g) \\ &= 2dyx_{los} + 2gy \end{aligned} \quad (38)$$

Lalu, dengan memasukkan persamaan x_{los} dan y_{los} kedalam persamaan α_k . maka:

$$\begin{aligned} (1 + d^2)x_{los}^2 + 2(dg - dy - x)x_{los} \\ + (x^2 + y^2 + g^2 - 2gy - R^2) = 0 \end{aligned} \quad (39)$$

merupakan standar penyelesaian secara analitik persamaan orde dua. Kemudian, dinotasikan lagi dengan:

$$a := 1 + d^2 \quad (39)$$

$$b := 2(dg - dy - x) \quad (40)$$

$$c := x^2 + y^2 + g^2 - 2gy - R^2 \quad (41)$$

maka solusi dari persamaan ini menjadi:

$$x_{los} = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a} \quad (42)$$

dimana: Jika $\Delta x > 0$, maka :

$$x_{los} = \frac{-b + \sqrt{b^2 - 4ac}}{2a} \quad (43)$$

dan jika $\Delta x < 0$, maka:

$$x_{los} = \frac{-b - \sqrt{b^2 - 4ac}}{2a} \quad (44)$$

Setelah x_{los} dihitung, maka y_{los} dapat dengan mudah dihitung. Akan tetapi jika $\Delta y = 0$, maka $y_{los} = y_k (= y_{k+1})$.

Argumen 2: jika $\Delta x = 0$, maka hanya persamaan (2.164) yang berlaku, dalam arti bahwa:

$$y_{los} = y \pm \sqrt{R^2 - (x_{los} - x)^2} \quad (45)$$

dimana $x_{los} = x_k (= x_{k+1})$. Perhitungan persamaan (45) melalui syarat :

Jika $\Delta y > 0$, maka:

$$y_{los} = y + \sqrt{R^2 - (x_{los} - x)^2} \quad (46)$$

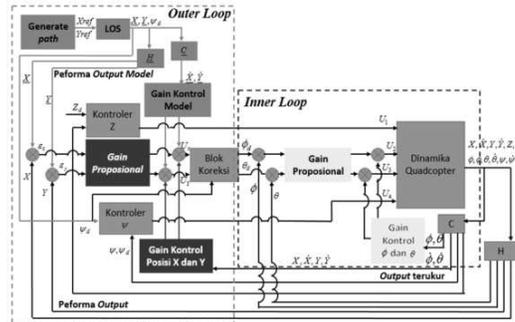
dan jika $\Delta y < 0$, maka:

$$y_{los} = y - \sqrt{R^2 - (x_{los} - x)^2} \quad (47)$$

ketika $\Delta x = 0, \Delta y = 0$, maka tidak ada pilihan berdasarkan [2], namun saran penulis bisa dimodifikasi dengan program.

2.4. Diagram blok quadcopter

Gambar 2 merupakan diagram blok dari strategi kontrol Quadcopter yang akan dirancang. Struktur kontrol terdiri model *following*, *inner* dan *outer loop*. Model *following* (yang ditandai dengan blok berwarna hijau) merupakan referensi x dan y model dari *generate path* serta penggunaan algoritma LOS dengan output berupa sudut ψ_d yang akan masuk dalam blok kontrol ψ , *Inner loop* merupakan kontroler rotasi sudut (ϕ, θ, ψ) yang menambahkan peforma H^∞ (ditandai dengan blok berwarna kuning), sedangkan *outer loop* merupakan kontroler *tracking* posisi (x, y, z) yang menggunakan struktur CGT (ditandai dengan blok berwarna merah). Sinyal referensi untuk sistem kontrol quadcopter ini terdiri atas referensi untuk posisi x_d, y_d, z_d dan sudut ψ_d .



Gambar 2. Gambar Diagram blok kontroler quadcopter

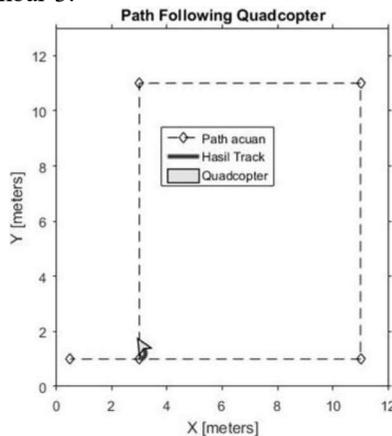
2.5. Hasil dan Pembahasan

Pengujian *path following* pada quadcopter secara otomatis dilakukan melalui simulasi, dengan adanya perubahan kecepatan angular. Kecepatan linear = 3,3, kecepatan sudut maksimal = 10, jarak arah hadap = 0,3 dan radius = 0,3. Titik yang akan dilalui untuk melakukan *path following* adalah sebagai berikut:

Tabel 2. Koordinat *Way point* ¹⁾

No	Titik (<i>Way Point</i>)	Koordinat
1	Titik 1	(3, 1)
2	Titik 2	(3, 11)
3	Titik 3	(11, 11)
4	Titik 4	(11, 1)
5	Titik 5	(0.5, 1)

Titik tersebut akan membentuk sebuah *path* (meter), adapun ilustrasi *path* yang akan dilalui oleh quadcopter ditunjukkan pada gambar 3.

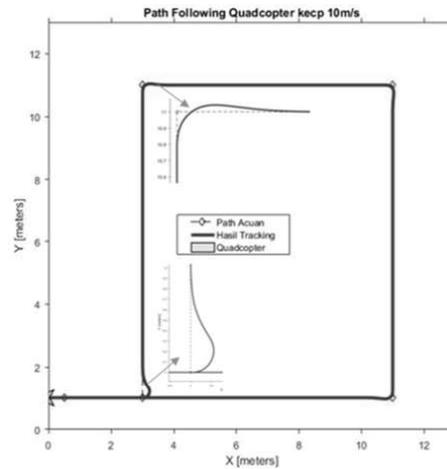


Gambar 3. Gambar *path* yang terdefinisi.

Dilakukan 2 pengujian *path following* menggunakan algoritma LOS (*Line of Sight*), yaitu pertama pengujian *path following* dengan kecepatan sudut 10m/s dan 20m/s.

2.1.1. Pengujian Path Following dengan kecepatan sudut 10m/s

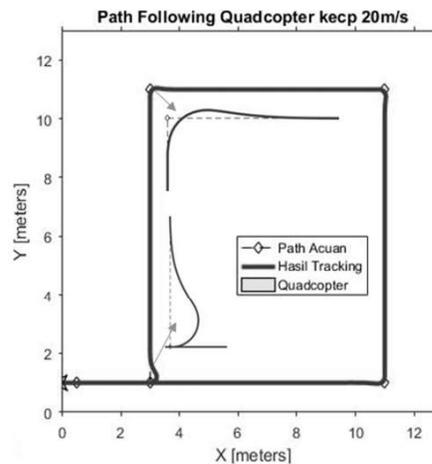
Pengujian gerak quadcopter dalam mengikuti suatu *path* (*path following*) tanpa adanya gangguan eksternal diilustrasikan pada gambar 4 dengan nilai *error* rata-rata posisi quadcopter terhadap *path* sebesar 0.8%.



Gambar 4. Gambar respon quadcopter terhadap *path* yang sudah terdefinisi dengan kecepatan 10m/s.

2.1.2. Pengujian Path Following dengan kecepatan sudut 20m/s

Pengujian gerak quadcopter dalam mengikuti suatu *path* (*path following*) tanpa adanya gangguan eksternal diilustrasikan pada gambar 5 dengan nilai *error* rata-rata posisi quadcopter terhadap *path* sebesar 1.2%.



Gambar 5. Gambar respon quadcopter terhadap *path* yang sudah terdefinisi dengan kecepatan 20m/s.

3. Simpulan

Algoritma LOS yang diterapkan pada quadcopter bekerja dengan baik ketika tidak ada gangguan eksternal dan mampu mengikuti jalur yang sudah didefinisi yang berupa *square* dan adanya perubahan kecepatan angular. Algoritma LOS mampu mengendalikan pergerakan quadcopter untuk melalui suatu *path* yang telah ditentukan dengan rata-rata *cross track error* kecil yaitu 0.8 meter dengan kecepatan 10m/s.

4. Saran

Saran untuk penelitian selanjutnya dapat membahas pengujian dengan adanya gangguan eksternal pada quadcopter dalam melakukan *steering*, kontrol kecepatan dalam melakukan maneuver, dan penambahan desain *observer* untuk mengestimasi seluruh *state* agar mendekati dengan kondisi *real*.

Daftar Pustaka

- [1]. David Cabecian, Rita Cunha, and Carlos Silvestre., “*A Globally Stabilizing Path Following Controller for Rotorcraft With Wind Disturbance Rejection*”, IEEE Transactions on Control Systems Technology Volume 23 issue 2, 2015.
- [2]. Thor I. Fossen., “Handbook of Marine Craft Hydrodynamics and Motion Control”, John Wiley & Sons, Inc., Canada, 2011.
- [3]. Quanser Q-Ball X-4 User Manual, Quanser Innovate Educate, 2010, Doc. 888 Rev.2 pp. i-46.