

Kapal Tak Berawak Menggunakan Andorid Untuk Deteksi Lintasan dengan *Color Tracking*

Koko Joni¹ dan Achmad Fiqi Abadillah¹

ABSTRACT : *Vista Bahari Kencana is an autonomous unmanned surface vessel that created to pass through the S shaped trajectory by tracking the buoy color. The V shaped vessel's hull is made from fiber glass because of its lightness and strength. It has 90 cm length, 30 cm width, 22.5 height and 6 kg weight. To track the buoy color, it uses Android camera and the processed data is sent to Arduino module through bluetooth communication. Arduino module processes the data to control a brushless DC motor and servo motor. To achieve the color tracking task, we take an advantage of OpenCV library. The tracking time along S shaped trajectory is around 37 seconds.*

Keywords: *autonomus color tracking, OpenCV, bluetooth, autonomus*

ABSTRAK : Vista Bahari Kencana merupakan kapal tak berawak jenis *autonomously* yang di buat untuk dapat melewati lintasan dengan pembacaan warna pada lintasan. Lambung kapal berbentuk V dengan bahan dasar *fiber glass* yang memiliki karakteristik ringan dan kuat. Kapal ini mempunyai panjang 90 cm, lebar 30 cm, tinggi 22,5 cm dan berat 6 kg. Untuk membaca lintasan, kapal menggunakan kamera Android dan hasilnya dikirim ke Arduino dengan *bluetooth*. Arduino mengolah sinyal yang didapat untuk menggerakkan motor dan servo. Aplikasi pengolahan citra menggunakan OpenCV dengan metode *color tracking*. Waktu yang dibutuhkan untuk melewati lintasan berbentuk S sekitar 37 detik.

Kata Kunci : *Autonomous, color tracking, openCV, Android*

PENDAHULUAN

Kendaraan tak berawak (*unmanned vehicle*) merupakan kendaraan yang tidak dikemudikan langsung oleh manusia, tetapi bisa menggunakan *remote* atau secara *autonomous*. Kendaraan yang tidak dipegang langsung oleh manusia sekarang ini banyak digunakan dan dikembangkan oleh berbagai negara untuk berbagai keperluan serta dibagi menjadi berbagai bentuk yaitu pesawat tak berawak untuk di udara (*Unmanned Aerial Vehicle/UAV*), di darat (*Unmanned Ground Vehicle/UGV*), di laut (*Unmanned Surface Vessels/USVs*) maupun di luar angkasa.

Negara maju seperti Amerika [1] dan Rusia [2] berlomba-lomba untuk mengembangkan teknologi ini, khususnya di bidang militer. Berbagai penelitian juga dilakukan untuk mengembangkan teknologi ini seperti tentang pengaturan motor penggeraknya menggunakan sistem hirarki [3], bagaimana membuat pesawat UAV ini mempunyai daya yang kuat karena digunakan untuk pemantauan dalam waktu yang lama sehingga diperlukan pengisian baterai dari jarak jauh menggunakan sistem laser [4]. Kapal tanpa awak (*USVs*) telah dikembangkan untuk berbagai keperluan. Kapal tanpa awak autonomus (*Autonomous Surface Craft/USCs*) telah dikembangkan untuk keperluan seperti militer, sipil maupun penelitian [5].

Teknologi untuk kapal tak berawak autonomus telah banyak dikembangkan, dimulai dari penggunaan sensor ultrasonik, GPS maupun pengolahan citra dengan kamera [6]. Kapal tanpa awak Vista Bahari Kencana (VBK) dibuat untuk keperluan Kontes Kapal Cepat Tak Berawak Nasional 2014 kategori Autonomus. Kapal ini dibuat untuk bisa melintasi lintasan yang berbentuk S dan bisa bermanuver membentuk lintasan U tanpa menyentuh pembatas yang berbentuk bola yang berwarna merah atau hijau. Pengolahan citra menggunakan kamera android hasilnya dikirim melalui *bluetooth* dan sinyal di proses dengan mikrokontroler Arduino untuk menggerakkan motor dan daun kemudi.

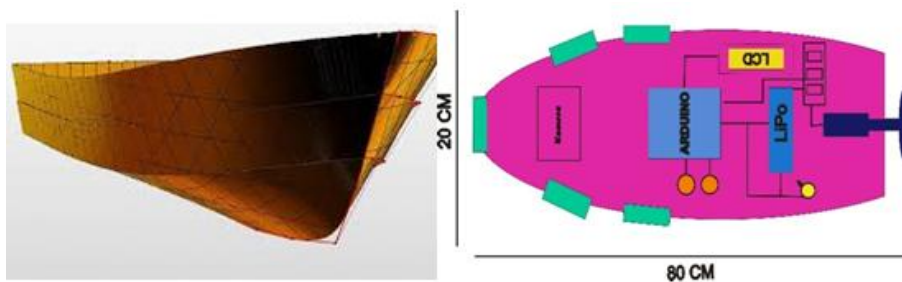
DESAIN KAPAL

Desain lambung kapal VBK mengacu pada salah satu kapal perang milik Indonesia yaitu KRI Banjarmasin. Bentuk lambung yang digunakan adalah lambung tunggal bentuk V (Gambar 1), di mana kapal dengan lambung lancip seperti huruf V mempunyai hambatan yang kecil terhadap air sehingga meminimalisir kemungkinan tenggelam. Ujung kapal VBK berbentuk pemecah ombak yang berfungsi agar kapal tidak goyah ketika diterpa ombak. Lambung tunggal V pada suatu kapal mempunyai beberapa keuntungan di antaranya mempunyai hambatan yang kecil terhadap air, di mana semakin kecilnya suatu hambatan terhadap gaya gesek antara kapal dan air maka akan mempengaruhi kecepatan kapal tersebut [7].

Kapal VBK menggunakan sebuah sumber daya, yaitu aki dan baterai Lippo. Untuk sistem pendorong, VBK Kapal menggunakan satu *Motor Brushless 750 KV* yang dilengkapi dengan *ESC 20A* sebagai pengendalinya. Kapal ini menggunakan *propeller* dengan tiga mata pisau untuk sistem pendorongnya (Gambar 2). *Propeller* dengan tiga mata pisau memiliki daya dorong yang kuat meskipun rpm pada motor kecil [8].

Untuk penempatan *rudder* (kendali kapal) diletakkan sejajar dengan *center line* (Gambar 3). Mengacu pada penelitian sebelumnya yang menyatakan bahwa semakin jauh *propeller* dari *center line*, maka waktu yang dibutuhkan kapal untuk berbelok juga semakin lama. Hal ini tepat untuk kapal *autonomous* agar pada saat berbelok pembacaan sensor pada rintangan menjadi maksimal [9].

¹ Jurusan Teknik Elektro Universitas Trunojoyo Madura Jawa Timur



■ Gambar 1. Desain Kapal



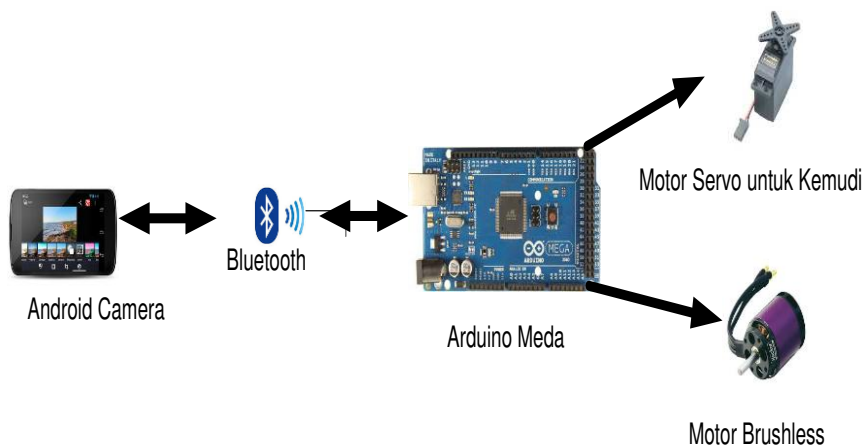
■ Gambar 2. Propeller 3 mata pisau



■ Gambar 3. Posisi Rudder

SISTEM KONTROL KAPAL

Pengendalian kapal VBK menggunakan kamera dan perangkat Adroid, kemudian pengiriman data dengan *bluetooth*, sebuah Arduino, motor servo untuk daun kemudi dan sebuah motor *brushless* untuk menggerakkan kapal (Gambar 4).



■ Gambar 4. Sistem Kontrol Kapal

Pendeteksi lintasan menggunakan kamera android, kemudian hasil pengambilan gambar diolah di HP android. Hasil pendeteksian menggunakan *color tracking* dikirim melalui *bluetooth* untuk diolah di Arduino yang digunakan untuk mengatur kecepatan kapal dan arah kemudi kapal. Android yang digunakan versi 4.2 Jelly Bean dengan spesifikasi sebagai berikut :

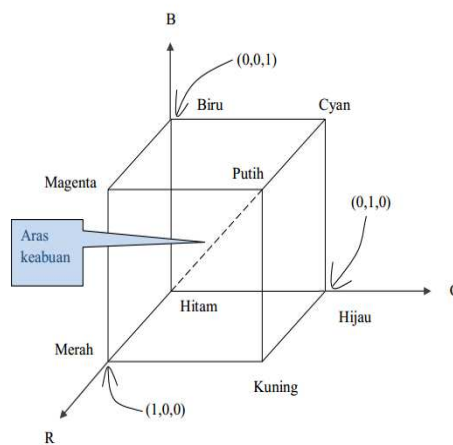
Arduino Mega berfungsi untuk mengendalikan servo sesuai dengan inputan dari kamera android yang dikomunikasikan melalui *bluetooth*. Posisi derajat tuas servo diatur dengan menggunakan kendali waktu pulsa berdasarkan perbandingan antara pulsa *high* dan *low*. Perbandingan dilakukan dengan menggunakan pengolahan citra. Pengolahan citra menggunakan kamera Android melalui OpenCV [10].

■ **Tabel 1.** Spesifikasi Android Evercross A5C

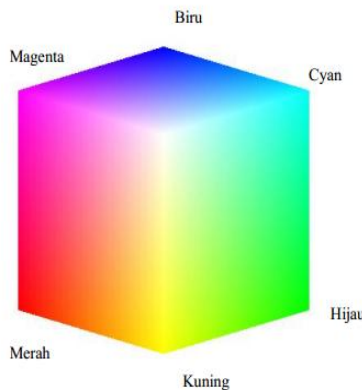
Hardware	Keterangan
Prosesor	Dual Core 1GHz
Memori Internal	256 Mb
RAM	512 Mb
Kamera	2 Mp
Konektifitas	Bluetooth, Wi-Fi, Wireless
Sistem Operasi	Android Jelly Bean 4.2

CITRA RGB

Pada dasarnya cahaya mempunyai tiga buah warna dasar, yaitu: merah (*red*), hijau (*green*), dan biru (*blue*). Hasil dari gabungan warna dasar tersebut dapat menghasilkan warna-warna lain. Untuk penyimpanan warna pada suatu pixel, unsur warna dasar cahaya disimpan dalam bentuk variable tipe long integer. Di mana setiap jenis warna menempati tempat sebesar 8 bit dengan kisaran untuk masing-masing warna red, green, dan blue antara 0-255 (Gambar 5). Maka kombinasi warna yang dihasilkan adalah $255*255*255$ atau sebesar 1677726 jenis warna (Gambar 6).



■ **Gambar 5.** Skema ruang warna RGB dalam bentuk kubus



■ **Gambar 6.** Kubus warna dengan 24 bit

CITRA HSV

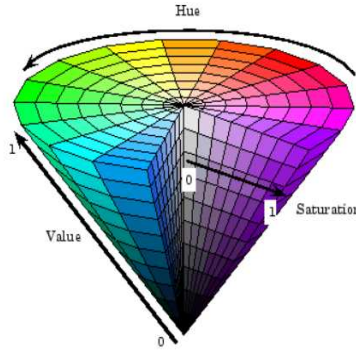
HSV merupakan contoh ruang warna yang merepresentasikan warna seperti yang dilihat oleh mata manusia. H berasal dari kata “hue”, S berasal dari “saturation”, L berasal dari kata “luminance”, I berasal dari kata “intensity”, dan V berasal dari “value” (Gambar 7).

Model HSV, yang pertama kali diperkenalkan A. R. Smith pada tahun 1978, ditunjukkan pada Gambar 8. Untuk mendapatkan nilai H, S, V berdasarkan R, G, dan B, terdapat beberapa cara. Cara yang tersederhana [11] adalah seperti berikut:

$$H = \tan\left(\frac{3(G-B)}{(R-G)+(R-B)}\right) \dots \dots \dots (1)$$

$$S = 1 - \frac{\min(R,G,B)}{V} \dots \dots \dots (2)$$

$$V = \frac{R+G+B}{3} \dots\dots\dots(3)$$



■ Gambar 7. Ruang warna HSV

Namun, cara ini membuat *hue* tidak terdefiniskan kalau S bernilai nol. Cara kedua menggunakan rumus-rumus yang digunakan sebagai berikut:

$$r = \frac{R}{(R+G+B)}, g = \frac{G}{(R+G+B)}, b = \frac{B}{(R+G+B)} \dots\dots\dots(4)$$

$$V = \max(r, g, b)$$

$$S = \begin{cases} 0, & \text{jika } V = 0 \\ 1 - \frac{\min(r,g,b)}{V}, & V > 0 \end{cases} \dots\dots\dots(5)$$

$$H = \begin{cases} 0, & \text{jika } S = 0 \\ \frac{60 * (g-b)}{S * V}, & \text{jika } V = r \\ 60 * \left[2 + \frac{b-r}{S * V} \right], & \text{jika } V = g \\ 60 * \left[4 + \frac{r-g}{S * V} \right], & \text{jika } V = b \end{cases} \dots\dots\dots(6)$$

$$H = H + 360 \text{ jika } H < 0 \dots\dots\dots(7)$$

MOMEN SPASIAL DAN MOMEN PUSAT

Momen spasial orde (m,n) didefinisikan sebagai berikut:

$$M_{ij} = \sum_{x=1}^M \sum_{y=1}^N x^i y^j I(x, y) \dots\dots\dots(8)$$

Dalam hal ini,

- a) i, j = 0, 1, 2, ..., dengan i j menyatakan orde momen;
- b) M menyatakan jumlah kolom pada citra;
- c) N menyatakan jumlah baris pada citra;
- d) x adalah ordinat piksel;
- e) y adalah absis piksel;
- f) I(x,y) menyatakan intensitas piksel pada posisi (x,y).

Adapun momen pusat adalah momen spasial yang dihitung relatif terhadap pusat massa. Jika pusat massa adalah (\bar{y}, \bar{x}) , momen pusat ditulis seperti berikut:

$$\mu_{ij} = \sum_{x=1}^M \sum_{y=1}^N (x - \hat{x})^i (y - \hat{y})^j I(x, y) \dots\dots\dots(9)$$

Momen di atas bersifat *invariant* (tidak terpengaruh) terhadap translasi. Dalam hal ini, \hat{x} dan \hat{y} diperoleh melalui:

$$\hat{x} = \frac{M_{10}}{M_{00}}, \hat{y} = \frac{M_{01}}{M_{00}} \dots\dots\dots(10)$$

Agar momen pusat bersifat bebas terhadap translasi, penyekalaan, dan rotasi, maka momen perlu dinormalisasi. Momen pusat ternormalisasi berupa:

$$\eta_{ij} = \frac{\mu_{pq}}{\mu^{\gamma}_{00}}, \gamma = \frac{i+j+2}{2} \dots\dots\dots(11)$$

MOMEN JARAK KE PUSAT

Apabila objek berupa kontur saja (misalnya berupa bentuk suatu pulau), momen dapat dihitung melalui jarak titik pada kontur terhadap pusat massa [12]. (Dalam hal ini, momen ke-p untuk sederet jarak d(n) didefinisikan sebagai berikut:

$$m_p = \frac{1}{N} \sum_{i=1}^N |d(n)|^p \dots\dots\dots(12)$$

Momen pusat ke-p didefinisikan sebagai berikut:

$$M_p = \frac{1}{N} \sum_{i=1}^N |d(n) - m_1|^p \dots\dots\dots(13)$$

Selanjutnya, momen-momen yang ternormalisasi didefinisikan sebagai:

$$m'_p = \frac{m_p}{(M_2)^{p/2}} \dots\dots\dots(14)$$

$$M'_p = \frac{M_p}{(M_2)^{p/2}} \dots\dots\dots(15)$$

Menurut Gupta dan Srinath [12] momen ternormalisasi m'_p dan M'_p bersifat bebas terhadap penyekalaan, rotasi, maupun translasi. Namun, mengingat momen berorde tinggi peka terhadap derau, mereka menganjurkan tiga fitur yang didefinisikan seperti berikut:

$$F_1 = \frac{\sqrt{M_2}}{m_1} = \frac{\sqrt{\frac{1}{N} \sum_{i=1}^N |d(n) - m_1|^2}}{\frac{1}{N} \sum_{i=1}^N d(n)} \dots\dots\dots(16)$$

$$F_2 = \frac{M_3}{(M_2)^{3/2}} = \frac{\frac{1}{N} \sum_{i=1}^N |d(n) - m_1|^3}{\left\{ \frac{1}{N} \sum_{i=1}^N |d(n) - m_1|^2 \right\}^{3/2}} \dots\dots\dots(17)$$

$$F_3 = \frac{M_4}{(M_2)^2} = \frac{\frac{1}{N} \sum_{i=1}^N |d(n) - m_1|^4}{\left\{ \frac{1}{N} \sum_{i=1}^N |d(n) - m_1|^2 \right\}^2} \dots\dots\dots(18)$$

Menurut Rangayyan [12], studi yang dilakukan oleh Shen, dkk. menunjukkan bahwa F2 dan F3 tidak menunjukkan perbedaan yang signifikan untuk bentuk yang berbeda dan F2 ternyata berubah secara nyata terhadap penyekalaan dan rotasi. Oleh karena itu, mereka melakukan modifikasi terhadap F2 dan F3 sebagai berikut:

$$F_2' = \frac{\sqrt[3]{M_3}}{m_1} = \frac{\sqrt[3]{\frac{1}{N} \sum_{i=1}^N |d(n) - m_1|^3}}{\frac{1}{N} \sum_{i=1}^N d(n)} \dots\dots\dots(19)$$

$$F_3' = \frac{\sqrt[4]{M_4}}{m_1} = \frac{\sqrt[4]{\frac{1}{N} \sum_{i=1}^N |d(n) - m_1|^4}}{\frac{1}{N} \sum_{i=1}^N d(n)} \dots\dots\dots(20)$$

Fakta yang menarik, kumpulan fitur $\{F1, F2', F3'\}$ mempunyai sifat-sifat seperti berikut [12]:

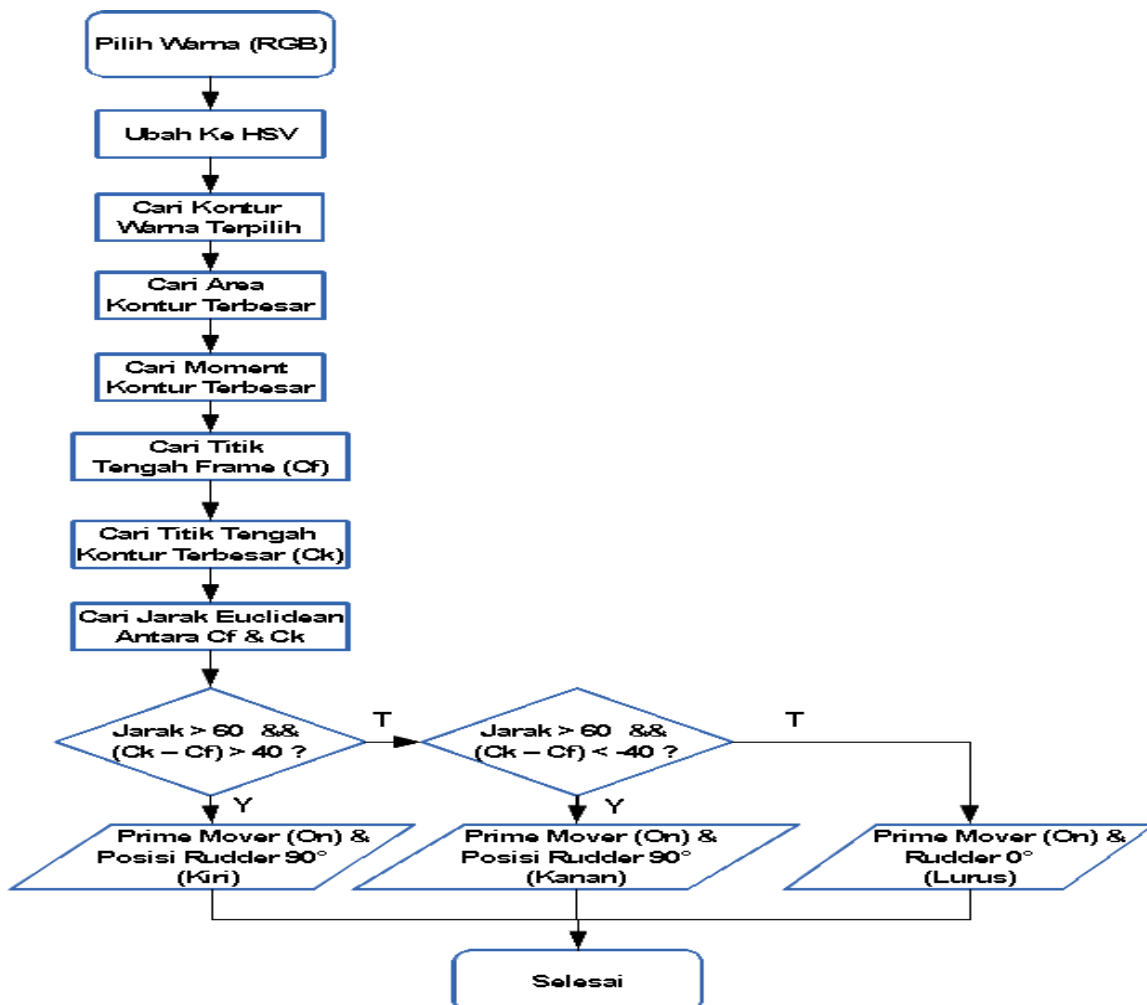
- Semua fitur dapat diperbandingkan secara langsung.
- $F3'$ menyatakan kekasaran kontur yang lebih baik daripada $F3$. Semakin besar nilai, semakin besar kekasaran kontur.

Temuan yang juga menarik, menurut Rangayyan, kombinasi $mf = F3' - F1$ merupakan indikator yang bagus untuk mengungkapkan kekasaran bentuk.

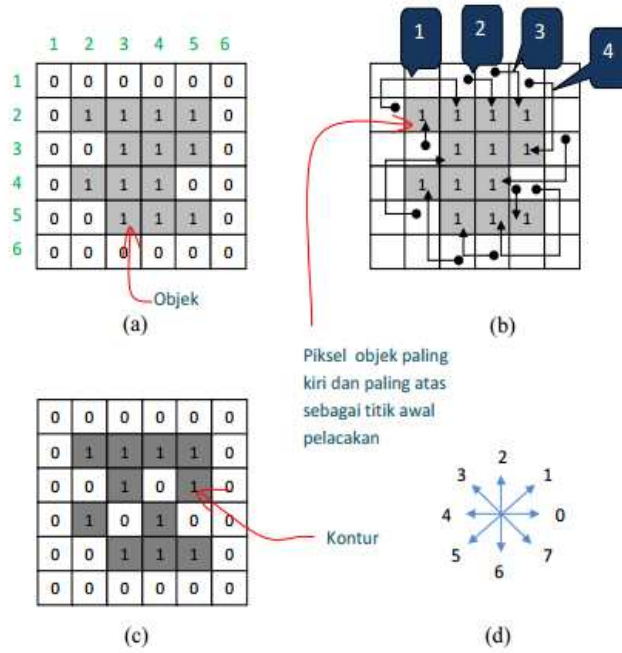
KONTUR INTERNAL

Salah satu cara untuk mendapatkan kontur internal yang telah diurutkan menurut letak piksel, yaitu dengan memanfaatkan algoritma pelacakan kontur Moore. Algoritma ini antara lain digunakan pada peta topografik digital [13]. Berikut adalah algoritma untuk memperoleh kontur internal dengan pelacakan kontur Moore (Gambar 8).

Algoritma akan membuat indeks pertama dan indeks terakhir pada kontur berisi nilai yang sama yaitu $b0$. Jika dikehendaki untuk tidak menyertakan nilai yang sama pada bagian akhir larik kontur, elemen tersebut tinggal diabaikan saja. Untuk memahami proses kerja pada algoritma Moore, perhatikan Gambar 9. Gambar 9(a) menyatakan keadaan objek pada citra. Piksel yang bernilai 1 menyatakan bagian objek dan yang bernilai 0 adalah bagian latarbelakang. Pada contoh tersebut, pelacakan akan dimulai pada posisi (2,2), yaitu piksel bagian objek yang terletak paling kiri dan paling atas. Adapun titik pencarian untuk piksel kedua dimulai di arah barat atau arah kiri piksel (2,2) tersebut. Pencarian dilakukan searah jarum jam. Pada langkah pertama, diperoleh piksel pada posisi (2,3). Pencarian berikutnya akan dimulai di posisi (1,3), yaitu yang ditandai dengan bulatan. Pada pencarian kedua, piksel yang didapat, yaitu posisi (2,4), dengan titik pencarian berikutnya dimulai di posisi (1,4). Pada pencarian ketiga, piksel yang didapat adalah pada (2,5), dengan titik pencarian berikutnya dimulai di posisi (1,5). Pada pencarian keempat, piksel yang didapat yaitu (3,5), dengan titik pencarian berikutnya dimulai di posisi (3,6). Jika langkah seperti itu terus diulang, suatu ketika akan diperoleh piksel yang sama dengan piksel yang pertama kali menjadi bagian kontur. Saat itulah proses untuk melacak kontur diakhiri. Semua langkah yang terjadi untuk contoh Gambar 9(a) ditunjukkan pada Gambar 9(b). Penomoran arah pencarian ditunjukkan pada Gambar 8.8(d), sedangkan hasil kontur diperlihatkan pada Gambar 9(c).



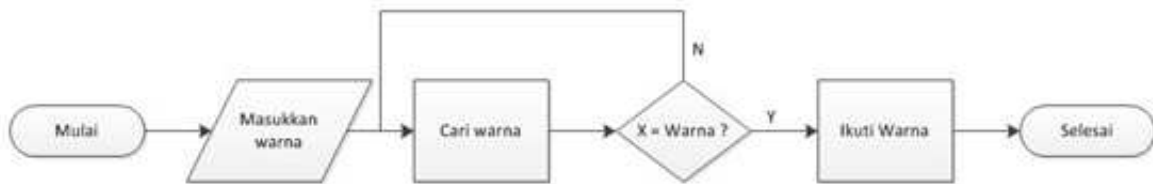
■ Gambar 8. Algoritma kontur Moore



■ Gambar 9. Penjelasan pelacakan kontur dengan menggunakan Algoritma Moore

COLOR TRACKING

Color tracking merupakan metode yang paling cocok untuk menyelesaikan misi yang akan diberikan pada saat kompetisi (Gambar 10). Pada metode ini user akan mengatur warna apa yang akan diikuti, kemudian kamera akan mencari keberadaan warna tersebut [14].



■ Gambar 10. Diagram Alir Color Tracking

Arduino berfungsi untuk mengatur Pulse Width Modulation (PWM) pada driver motor sehingga berpengaruh pada putaran motor brushless. Semakin renggang PWM, maka akan semakin lambat perputaran motor, dan semakin sempit PWM, maka semakin cepat perputaran motor.

HASIL DAN PEMBAHASAN

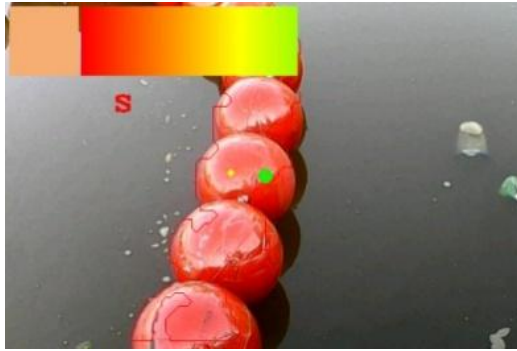
Kapal tak berawak autonomous menggunakan desain lambung tunggal berbentuk V (Gambar 11). Keuntungan dari bentuk lambung ini adalah kestabilan kapal bisa terjaga, serta maneuver kapal menjadi maksimal. Disamping itu, dengan menggunakan lambung V, maka keseimbangan kapal menjadi lebih baik.



■ Gambar 11. Kapal Vista Bahari Kencana

Daun kemudi kapal VBK menggunakan satu daun kemudi (rudder), dimana satu daun kemudi ini dapat menghasilkan suatu belokan yang presisi dan optimal. Pada awalnya penempatan rudder berada di sebelah kanan dari propeller. Hal ini menyebabkan waktu yang dibutuhkan kapal untuk berbelok terlalu cepat sehingga menyulitkan sensor untuk mendeteksi rintangan. Kemudian penempatan rudder dirubah menjadi tepat di atas propeller, ternyata waktu yang dibutuhkan kapal untuk berbelok sedikit lebih lama, hal ini menyebabkan sensor pada kapal dapat mendeteksi rintangan lebih baik.

Sumber tegangan pada VBK menggunakan Aki 12V yang disambungkan ke *Arduino Mega ADK*. Dengan adanya penambahan aki kering juga menambah beban kapal mengingat sebelum ditambahkan aki kering kapal VBK terlalu ringan sehingga *propeller* tidak bisa masuk ke air secara keseluruhan sehingga mengurangi daya dorong *propeller*.



■ Gambar 12. Hasil pembacaan Color Tracking

Pengolahan citra dengan *Color tracking* menggunakan kamera android (Gambar 12). Metode *color tracking* dipilih karena cepat dalam pengolahan menggunakan telepon genggam berbasis android dengan spesifikasi yang tidak terlalu tinggi. Kamera android menangkap objek lintasan yang berwarna dengan *color tracking* hasil pembacaan kemudian di kirim melalui *bluetooth* dan diterima di *Arduino* untuk menterjemahkan sinyal yang digunakan untuk menggerakkan daun kemudi. Hasil pembacaan kamera dan posisi daun kemudi bisa di baca pada Tabel 2.

■ Tabel 2. Hasil *image processing color tracking* terhadap *rudder*

<i>Ueclidean distance</i>	$CoG (y) - im_center (y) (r)$	Perintah ke mikro	Respon <i>rudder</i>	Sudut respon
>60	>40	Y	Kiri	(+) 90°
>60	<-40	X	Kanan	(-) 90°
>0	$40 \leq c \leq -40$	S	Lurus	0°

Center of Gravity (CoG) merupakan titik tengah dari target yang ditangkap oleh kamera. Sumbu yang diambil dari titik tengah ini adalah sumbu-y. Sedangkan *im_center* merupakan titik tengah dari kamera Android pada sumbu-y. Setelah dua titik ini ditemukan, maka akan dibandingkan selisih diantara jarak tersebut sehingga kita menemukan nilai dari *range (r)*. Percobaan yang dilakukan pada lintasan berbentuk S dengan jarak kurang lebih 10 meter ditempuh dalam waktu 37 detik, tanpa menyentuh lintasan yang berada di samping kiri maupun kanan.

KESIMPULAN

Kapal tak berawak menggunakan bentuk lambung V dan untuk menggunakan kamera Android 2 Mpx sudah bisa melakukan jejak lintasan dengan baik. Metode *color tracking* tepat digunakan untuk perangkat dengan spesifikasi tidak terlalu tinggi. Penempatan *rudder* yang menjauh dari *center line* memberikan respon yang tepat untuk perbelokan kapal sehingga tidak menyentuh lintasan. Untuk penelitian selanjutnya perlu di bentuk desain bentuk kapal yang lebih aerodinamis karena kapal ini rentan dengan angin. Untuk motor perlu di pilih motor yang sesuai untuk penggerak yang digunakan di dalam air. Untuk metode pengolahan citra juga perlu di pilih metode lain dan dengan perangkat yang lebih tinggi untuk menghadapi masalah dan rintangan yang lebih rumit.

DAFTAR PUSTAKA

- [1] US Army, "US Army Unmanned Aircraft System Roadmap 2010-2013", U.S ARMY, 2010
- [2] Gordon, Fefim, "Soviet/Russian Unmanned Aerial Vehicle", Midland, 2005
- [3] Shim, DH; Kim, H; Sastry, S, "Hierarchical Control System Synthesis For Rotorcraft-Based Unmanned Aerial Vehicle", AIAA Guidance, Navigation and Control Conference, 2000.
- [4] Nugent, TJ; Kare JT, 2005, "Laser Power For UAVs", A White Paper
- [5] J. E. Manley, "Unmanned surface vehicles, 15 years of development," in *Oceans*, 2008, pp. 1–4.
- [6] N.P. Papanikolopoulos, P.K. Khosla, and T. Kanade. Visual tracking of a moving target by a camera mounted on a robot: A combination of control and vision. *IEEE Trans. Robotics and Automation*, 9:14–35, 1993
- [7] Michelle, Howard. "Autonomy's Hugh Jackman" University of Michigan, Ann Arbor, MI 48109.
- [8] Geer, Dave. "The Propeller Handbook", 2001
- [9] Bradski, Gary, Kaebler Adrian "Learning OpenCV", 2008.

- [10] Bharathi, M. Amutha, "Color Detection In OpenCV".
- [11] Acharya, T.; Ray, A.K., "*Image Processing Principles and Applications*", New Jersey: John Wiley & Sons, Inc, 2005.
- [12] Rangayyan, R.M., "*Biomedical Image Analysis*. Boca Raton", CRC Press, 2005
- [13] Prada,R., Kumar, S.; Agarwal, R.; Pradhan, M.P.; and Ghose, M.K, "*Contour Line Tracing Algorithm for Digital Topographics Map*", International Journal of Image Processing (IJIP), Vol. 4, No. 2. . 2010.
- [14] Saragih-Rapelman,Adji-Surjo Widodo, Amiadji. "Pengaruh Jumlah dan Posisi Rudder Terhadap Kemampuan Manoeuvring Kapal". Institut Teknologi Sepuluh Nopember.