

# MULTITARGET TRACKING MENGGUNAKAN MULTIPLE HYPOTHESIS TRACKING DENGAN CLUSTERING TIME WINDOW DATA RADAR

Gigih P. Cahyono<sup>1)</sup>, Handayani Tjandrasa<sup>2)</sup>

<sup>1,2)</sup>Jurusan Teknik Informatika, Fakultas Teknologi Informasi  
Institut Teknologi Sepuluh Nopember, Surabaya  
email: [gigih.prasetyo18@gmail.com](mailto:gigih.prasetyo18@gmail.com)<sup>1)</sup>, [handatj@its.ac.id](mailto:handatj@its.ac.id)<sup>2)</sup>

## ABSTRAK

Sistem radar dibagi menjadi dua jenis yaitu sistem radar sipil dan sistem radar militer. Kedua jenis sistem radar tersebut memiliki kesamaan yaitu telah digunakan untuk pemantauan lalu lintas udara. Pesawat yang dipantau di udara saat ini mengalami jumlah peningkatan yang besar sehingga untuk memudahkan pemantauannya diperlukan suatu sistem yang dinamakan multitarget aircraft tracking. Penelitian ini bertujuan untuk mendapatkan algoritma multitarget tracking (MTT) yang valid, yaitu dengan menggunakan kombinasi preprocessing data radar dengan clustering time window (CTW) dan algoritma Multiple Hypothesis Tracking (MHT). Penelitian ini diawali dengan penyiapan data rekaman radar yang direkam langsung. Kemudian data rekaman tersebut disimulasikan dengan algoritma yang telah dirancang. Pengecekan kemampuan algoritma tersebut dilakukan dengan membandingkannya dengan MHT tanpa clustering time window. Uji coba dilakukan dengan memakai data rekaman berdurasi kurang lebih 18 menit. Uji coba tersebut menghasilkan nilai correct target sebesar 87.66%, undetected target sebesar 12.81%, maintain target sebesar 80.5% dan inexistent target sebesar 23.65%. Dari hasil uji coba menunjukkan bahwa metode yang diusulkan lebih bagus jika dibandingkan dengan metode MHT tanpa CTW.

**Kata kunci:** aircraft, clustering time window, Multiple Hypothesis Tracking, multitarget tracking, sistem radar.

## ABSTRACT

The radar systems are categorized into two types: civilian and military radar systems. The two types of radar systems have similarity that has been used for monitoring air traffic. The aircraft that is monitored in the air is currently experiencing a large amount improvement. To facilitate the aircraft monitoring, it required a system called multitarget aircraft tracking. This research proposed a hybrid method of multitarget tracking algorithm (MTT) with time window clustering for data pre processing. This research started with the preparation of the radar data recorded directly. Data recording was simulated with algorithms that have been designed. Finally, performance of the algorithm was compared with multiple hypothesis tracking without clustering time window. The test is done using data recording lasted approximately eighteen minutes. The test result shows that the proposed method is better than the MHT without CTW method. It can be seen from the correct target 87.66%, the undetected target 12.81%, the maintain target 80.5% and the inexistent target 23.65%.

**Keywords:** aircraft, clustering time window, Multiple Hypothesis Tracking, multitarget tracking, radar systems.

## I. PENDAHULUAN

Dalam lalu lintas udara, ada berbagai jenis pesawat dengan jumlah yang tidak sedikit. Untuk memantau lalu lintas tersebut diperlukan suatu sistem yang disebut sistem radar. Selama ini sistem radar yang sudah ada dibagi menjadi dua jenis yaitu sistem radar sipil dan sistem radar militer. Kedua jenis sistem radar tersebut telah digunakan untuk pemantauan lalu lintas udara. Dalam proses pemantauan tersebut, telah diidentifikasi berbagai jenis pesawat misalnya pesawat sipil, militer, latih dan bahkan ada pesawat tak berawak. Karena banyaknya jenis pesawat tersebut, maka diperlukan suatu sistem yang mampu untuk mengidentifikasi setiap lintasan dari pesawat-pesawat tersebut sehingga memudahkan proses pemantauannya. Proses pengidentifikasian tersebut biasa disebut dengan proses *Aircraft Tracking*.

Penelitian tentang *tracking* yang cukup kompleks yaitu mengatasi proses manuver pesawat yang dilakukan oleh Pineda dkk [1]. Pada penelitian ini, proses *tracking* dilakukan pada pesawat tempur dalam kondisi beberapa manuver antara lain *dogfight* dan pesawat tempur dalam kondisi sedang berhadapan dengan musuhnya. Kemudian untuk proses pengolahan data *tracking*, penelitian yang dilakukan Strauss dkk [2] menggunakan *spatio-temporal clustering* tetapi yang diolah bukan data yang diambil langsung dari radar.

*Aircraft Tracking* ini juga sudah dilakukan penelitian oleh beberapa peneliti sebelumnya. Proses *aircraft tracking* secara khusus berdasarkan penelitian yang dilakukan sebelumnya dibedakan menjadi dua jenis, jika dilihat berdasarkan jumlah target pesawat, yaitu *single target tracking* (STT) dan *multi-target tracking* (MTT). Untuk STT, penelitian yang pernah dilakukan oleh Park, Chunki dkk menggunakan metode *Minimum Spanning Tree* dengan dipadukan dengan metode klustering dan proses pengolahan datanya dilakukan dengan cara *clustering time window* [3]. Untuk kategori MTT, lebih banyak peneliti yang mengembangkannya sampai saat ini. Beberapa

metode yang sudah dipakai untuk proses MTT ini yaitu *Probabilistic Data Association Filter* (PDAF), *Multiple Hypothesis Tracking* (MHT) dan *Interactive Multiple Model* (IMM). Metode-metode tersebut cukup baik ketika digunakan untuk proses *multi-target tracking*, misalnya penelitian yang dilakukan oleh Jiang, Xue dkk yang mengkombinasikan metode *Joint Maximum Likelihood Probabilistic Data Association* (JML-PDA) sebagai inisialisasi track dengan PDAF sebagai *track maintenance* [4]. Metode IMM dan MHT pernah dikombinasikan oleh Blackman [5]. Pada penelitian tersebut IMM dipakai untuk menentukan *track score* (nilai probabilitas) dari setiap hipotesis proses asosiasi dan *maintenance track* pada MHT. Berdasarkan beberapa penelitian tersebut MTT dengan MHT terlihat cukup baik ketika digunakan untuk proses *aircraft tracking* khususnya untuk sistem data *tracking* yang konflik. Selain fleksibel dikombinasikan dengan metode lain, metode ini juga sangat bagus dalam tahapan proses MTT yaitu proses *track initiation*, *plot association* dan *track maintenance* [5]. Proses MTT ini sangat penting nantinya untuk membantu *Air Traffic Controller* (ATC) dan pihak pemantau lalu lintas pesawat yang lain, misalnya dari pihak militer. MHT juga dikenal sebagai salah satu algoritma MTT yang cukup baik. Selain digunakan untuk proses *tracking* udara khususnya simulasi pesawat, MHT pernah juga digunakan untuk penelitian *ground tracking*, salah satunya pada proses *tracking* robot [6]. Proses *tracking* dengan metode MHT dan menggunakan proses kluster untuk pengelompokan data setiap langkah lintasan robot berdasarkan waktu. Pada proses algoritma MHT, terdapat langkah proses penentuan hipotesis dimana pada tahapan ini ada proses penentuan nilai *track score* atau nilai probabilitas yang cukup optimal yaitu algoritma Murty [7] dan [8] yang memang sering digunakan di dalam metode MHT. Pengelompokan data pada proses MHT juga pernah dilakukan oleh Makris dkk [9]. Mereka meneliti pengelompokan data berbasis *Mesoscale Convective System* (MCS). Tipikal sistem ini dideteksi dengan menerapkan ambang batas dari inframerah.

Secara garis besar beberapa proses yang dikerjakan pada proses MTT ini yaitu *track association* (TA), *track smoothing* (TS), *track initiation* (TI) dan *track maintenance* (TM) [10]. Istilah-istilah dalam proses MTT tersebut secara umum adalah sebagai berikut: TA merupakan suatu proses asosiasi plot dengan suatu track yang sudah terbentuk, TS adalah proses *update* suatu track dengan plot hasil asosiasi yang terakhir, TI merupakan proses pembentukan track baru dengan beberapa plot yang tidak terasosiasi dengan track yang sudah ada, sedangkan TM adalah suatu proses *delete* track yang sudah tidak mengalami *update* selama waktu tertentu. Dibandingkan dengan STT, MTT lebih sering dikembangkan untuk penelitian karena sistem MTT ini dalam praktek sehari-harinya sering digunakan, khususnya dalam dunia penerbangan yaitu sebagai sistem pemantauan penerbangan. Dalam sistem pemantau penerbangan tersebut, jarang ditemui *target* dalam hal ini pesawat yang terpantau berjumlah tunggal, yang lebih sering ditemui yaitu *target* tersebut berjumlah lebih dari satu atau *multi-target*.

Berdasarkan beberapa penelitian sebelumnya di atas, diperoleh beberapa informasi tentang *preprocessing* data radar dan beberapa metode MTT serta metode STT dengan penjelasan percobaan masing-masing yang menampilkan hasil yang cukup bagus. Dari hasil penelitian yang dilakukan oleh Xue Jiang dkk dan Blackman, diketahui ada dua algoritma MTT yang cukup bagus, yaitu JPDA dan MHT. Namun, dari dua algoritma tersebut, MHT lebih baik ketika menemui proses *tracking* pada data radar yang konflik [5], dimana kondisi data seperti itu sering muncul pada proses sistem data pada proses MTT. Berdasarkan informasi tersebut, peneliti mencoba melakukan penelitian tentang suatu proses MTT dengan menggabungkan metode *preprocessing* data radar [3] menggunakan *clustering time window* dengan salah satu metode MTT yaitu MHT. Penggunaan CTW pada MHT sangat penting karena pada proses tersebut data akan dikelompokkan berdasarkan satu putaran radar, sehingga pada kelompok data tersebut tidak akan ditemui data dua pesawat yang sama karena jika dua pesawat yang sama tersebut diproses pada MHT akan menyebabkan salah *tracking*. Hal itu terjadi karena pada MHT hanya bisa melakukan *update* setiap *track* sebanyak satu kali sehingga jika ada dua pesawat yang sama diproses, pasti salah satu pesawat akan membentuk lintasan sendiri atau bisa juga bergabung dengan lintasan pesawat lain yang dapat menyebabkan lintasan pada pesawat tersebut tidak *valid*.

Berdasarkan kasus tersebut, pada penelitian ini diusulkan suatu sistem MTT yaitu dengan MHT yang menggunakan praproses data radar menggunakan CTW. Dari kombinasi MHT-CTW tersebut, diharapkan bisa didapatkan hasil MTT yang *valid*, artinya lintasan masing-masing pesawat bisa menjadi *track* sesuai dengan lintasan yang sebenarnya. CTW berperan dalam pengolahan data radar dengan cara mengelompokkan data berdasarkan satu putaran radar, sedangkan MHT berperan sebagai algoritma MTT.

## II. DATA RADAR PESAWAT

Radar atau disebut *radio detection and ranging* adalah salah satu alat navigasi. Ada dua jenis radar yang ada di negara kita ini, yaitu radar sipil dan radar militer dimana fungsi dari keduanya tentu berbeda. Hampir setiap radar mempunyai dua macam tipe radar yaitu *Secondary Surveillance Radar* (SSR) dan *Primary Surveillance Radar* (PSR), dimana SSR adalah radar yang akan menangkap sinyal-sinyal pesawat yang mengirim ID melalui

*transponder* pesawat, sedangkan PSR adalah radar yang mampu menangkap sinyal-sinyal pesawat meskipun pesawat tidak mengirimkan ID melalui *transponder* nya.

Bentuk data untuk operasi algoritma pelacakan ada dua macam. Yang pertama berbentuk operasi koordinat kartesian variabel, yang dipakai adalah  $x$ ,  $y$  dan *altitude*. Bentuk yang kedua adalah operasi koordinat polar yang dihasilkan dari suatu sensor, variabel yang digunakan adalah *elevation*, *range* dan *bearing*. Pada penelitian ini, data yang digunakan adalah data hasil dari sensor, tetapi untuk operasi perhitungannya, akan menggunakan operasi koordinat kartesian sehingga nilai dari *range* dan *bearing* akan dikonversi menjadi nilai  $x$  dan  $y$ . Secara umum, yang dimaksud dari *range* ( $r$ ) dan *bearing* ( $\theta$ ),  $r$  merupakan jarak antara pusat radar dengan *target* dalam kasus ini adalah pesawat, sedangkan  $\theta$  adalah sudut yang dibentuk antara *target* dengan sumbu *true north* dari radar [11]. Untuk lebih jelasnya bisa diperhatikan pada Gambar 1.

### III. METODOLOGI

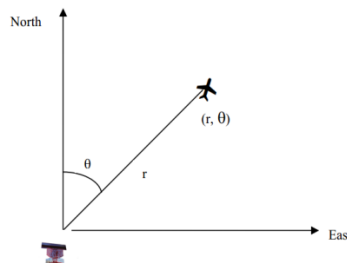
#### A. Data Rekaman Radar

Data rekaman radar yang digunakan adalah data radar dari salah satu radar di Indonesia. Data tersebut telah direkam hanya untuk proses penelitian tentang *tracking* pesawat. Data yang dihasilkan berupa informasi-informasi yang diperlukan untuk proses penelitian untuk *tracking* dari pesawat. Data-data tersebut meliputi waktu kedatangan data, *range*, *bearing*, *mode 3* dan *altitude*.

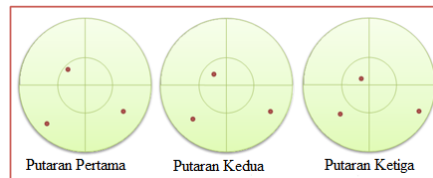
Waktu kedatangan data yaitu *time* atau waktu dimana data tersebut berhasil ditangkap oleh radar, *range* merupakan jarak antara pusat radar dengan *target* atau pesawat yang memiliki satuan *nautical mile* (Nmi). Kemudian *bearing* yaitu sudut yang dibentuk antara *target* dengan North-Signal Radar [12]. Berikutnya Mode 3 adalah identitas setiap pesawat dimana setiap pesawat memiliki nilai mode 3 yang berbeda-beda. Sedangkan *altitude* adalah ketinggian pesawat diukur dari atas permukaan laut atau *sea level* dan memiliki satuan *feet*. Dengan menggunakan data-data yang diperoleh langsung dari radar tersebut, akan dihitung juga *speed* atau kecepatan dan juga *heading*. Proses perhitungan keduanya akan diproses pada saat konversi data. Sebagai contoh untuk bentuk data rekaman tersebut adalah seperti TABEL I.

#### B. Clustering Time Window

*Clustering time window* merupakan suatu proses kluster data berdasarkan waktu dan mempunyai pengertian secara khusus bahwa data hasil rekaman akan dikelompokkan atau dikluster berdasarkan waktu satu kali putaran radar, yang berarti data akan diproses untuk *tracking* setelah selesai satu kali putaran radar. Berikut contoh tiga kali tangkapan satu putaran radar [3] seperti pada Gambar 2.



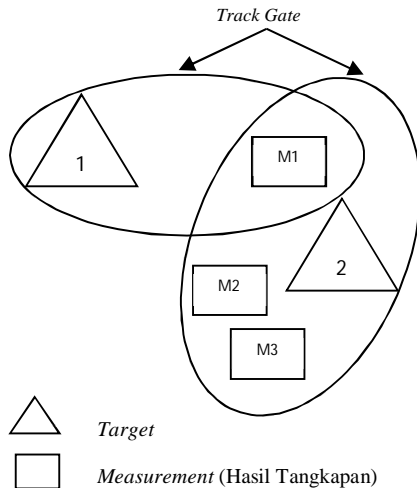
Gambar 1. Tata Acuan Koordinat Sistem Radar



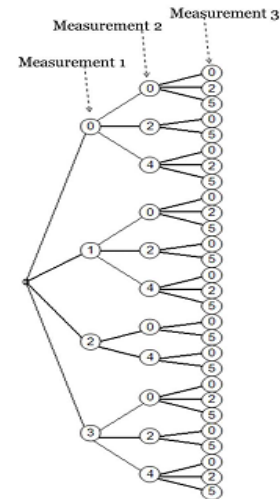
Gambar 2. Tangkapan Tiga Kali Putaran Radar

TABEL I  
REKAMAN DATA RADAR

Time	Range	Bearing	Mode3	Altitude
0:13:06	117	220.8352	902	220
0:13:06	21	235.2527	908	191
0:13:07	90	248.6154	903	314
0:13:07	98	277.0989	0	2
0:13:11	163	54.85714	1694	2
0:13:11	144	61.18681	3673	388
0:13:11	46	73.84615	1332	331



Gambar 3. Tipikal Data Asosiasi pada Keadaan Konflik



Gambar 4 Representasi Hipotesis yang Terbentuk

Pada kenyataannya beberapa radar memiliki waktu putaran yang berbeda-beda, tetapi pada umumnya lama putarannya adalah 6 rpm (radian per menit) atau sekitar 10 detik. Selama satu putaran tersebut sudah dapat dipastikan bahwa tidak mungkin ada pesawat yang ditangkap radar lebih dari satu kali atau *double*. Tetapi ada kemungkinan ketika pada tangkapan putaran pertama berhasil didapatkan tetapi pada tangkapan putaran kedua tidak berhasil ditangkap, kejadian seperti itu bisa terjadi karena kemampuan tangkapan dari suatu radar itu sendiri.

### C. Algoritma Reid

Algoritma ini berasal dari ide propagasi atau penyebaran beberapa hipotesis penugasan pada kasus *target* tunggal dalam kondisi *false alarm* (salah *target*) yang pertama diteliti oleh Singer [13]. Kemudian Reid [14] pertama mengembangkan sebuah algoritma yang lebih lengkap. Algoritma Reid mendefinisikan sebuah cara sistematis dimana beberapa data asosiasi hipotesis dapat dibentuk dan dievaluasi untuk kasus beberapa target di dalam kondisi *false alarm*.

Algoritma Reid berisi tiga komponen utama dalam penentuan hipotesisnya yaitu *track* baru, *false alarm* dan *track* yang sudah terbentuk. Berdasarkan Gambar 3, masing-masing komponen tersebut misalnya untuk *track* baru berarti merupakan hasil tangkapan yang baru (*measurement*) yaitu M1, M2 dan M3, kemudian untuk *false alarm* diberikan simbol 0 pada pembentukan hipotesisnya. Dan untuk *track* yang sudah ada berdasarkan Gambar 3 di atas dituliskan simbol angka 1 dan 2 yang berada di dalam simbol segitiga, sedangkan untuk penulisan M1, M2 dan M3 yang berada pada simbol kotak, dalam pembentukan hipotesis disimbolkan angka secara berturut-turut yaitu 3, 4 dan 5.

Gambar 4 merupakan hasil representasi hipotesis [15] yang terbentuk untuk kasus seperti pada Gambar 3. Berdasarkan hasil representasi tersebut, *measurement* pertama menghasilkan empat macam cabang yaitu *false alarm* 0, *target* 1, 2 dan *target* 3. Selanjutnya, untuk *measurement* kedua, setiap cabang terbentuk maksimal tiga cabang, yaitu *false alarm* 0, *target* 2 dan *target* 4 yang terbentuk dari hasil *measurement* kedua. Dan untuk *measurement* ketiga setiap cabang maksimal memiliki tiga macam target yaitu *false alarm* 0, *target* 2 dan *target* 5. Sebagai catatan pada proses MHT berdasarkan algoritma Reid ini, misalnya pada *measurement* pertama pada *target* 2 untuk proses *measurement* kedua yang seharusnya pada semua anggota setiap cabangnya mendapat *assignment target* 2 tidak akan dilakukan. Hal tersebut tidak dilakukan karena di dalam sebuah baris hipotesis tidak diperbolehkan membentuk hipotesis dengan *target* yang sama. Kemudian khusus untuk *target* yang sudah dipastikan menjadi sebuah *target*, misalkan dalam contoh ini adalah target 1 dan 2, bisa dijadikan kemungkinan pembentukan hipotesis pada *measurement* berikutnya dengan syarat *target* tersebut berada di dalam *gate* dari *measurement* pada saat itu. Pada kasus Gambar 2.3 misalnya, terjadi pada *target* 2. *Target* 2 tersebut selalu digunakan untuk membentuk hipotesis pada *measurement* kedua dan ketiga karena berada di dalam *gate* ketika terjadi *measurement* kedua dan ketiga. Pada hasil representasi tersebut terbentuk 28 macam hipotesis, dari hasil bentuk hipotesis seperti pohon bilangan tersebut, selanjutnya akan dibentuk dua dimensi *array* untuk proses komputerisasi. Bentuk tersebut bisa disebut matrik hipotesis yang mempunyai sebuah baris untuk setiap hipotesisnya dan sebuah kolom untuk proses tangkapan (*measurement*).

### D. Penentuan Nilai Probabilitas

Dari banyaknya nilai probabilitas dari setiap hipotesis yang dihasilkan berdasarkan algoritma Reid memang kurang begitu efektif. Terutama disebabkan terlalu banyaknya kemungkinan hipotesis yang dihasilkan. Maka berdasarkan penelitian yang dilakukan oleh Cox [7], algoritma Reid akan sangat efisien jika dalam implementasinya menggunakan algoritma Murty. Murty [12] efisien untuk Reid karena algoritma tersebut dapat menemukan *m-best* solusi untuk masalah penugasan (*assignment*). Penggunaan pendekatan ini yaitu dengan  $m_p(k - 1)$  hipotesis dari hasil tangkapan sebelumnya. Jumlah dari hipotesis yang terbentuk pada tangkapan saat sekarang dapat terbatas untuk  $m(k)$  ketika  $m$  adalah suatu input parameter yang dapat menjadi priori atau mungkin dipilih secara adaptif [5]. Dari algoritma Murty ini, diharapkan akan didapatkan nilai probabilitas yang terbesar yang berarti hipotesis dari pemilik nilai probabilitas tersebut merupakan hipotesis yang terbaik.

E. Algoritma Kalman Filter

Pada proses penentuan hipotesis, diperlukan penentuan estimasi atau prediksi target. Untuk menghitungnya, akan digunakan algoritma Kalman Filter [7]. Kalman Filter yang digunakan yaitu Kalman Filter linier yang sederhana dengan *state vector* =  $[x \ y \ \dot{x} \ \dot{y}]'$ , dimana  $x$  dan  $y$  adalah koordinat posisi dari *target*. Kemudian *state transition* diberikan nilai matrik F sebagai berikut.

$$F(k) = \begin{bmatrix} 1 & dt & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & dt \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Vektor *measurement*  $z(k) = [x \ y]'$  dan matrik observasi dari H sebagai berikut.

$$H(k) = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}$$

Dan untuk matrik *measurement noise* R digunakan matrik sebagai berikut.

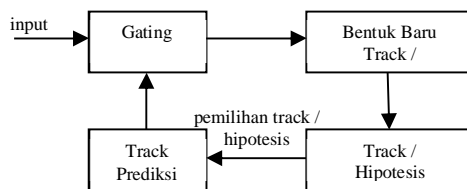
$$R(k) = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

Pada penelitian ini, akan digunakan algoritma Kalman Filter [16] untuk melakukan proses prediksinya.

F. Algoritma Multiple Hypothesis Tracking

Pada dasarnya algoritma MHT bisa digunakan untuk objek target yang memiliki lebih dari dua atau tiga dimensi [15]. Setiap hipotesis berisi target yang sudah terbentuk dan juga lintasan atau *track* dari target tersebut. Sistem ini nantinya akan menerima tangkapan secara periodik. Semua hasil tangkapan dinotasikan dengan  $Z^k$  pada waktu ke  $k$ , dan tangkapan ke  $l$  pada waktu ke  $k$  dinotasikan  $Z_l^k$ . Hasil tangkapan tersebut pada dasarnya berisi suatu posisi baik dua dimensi atau tiga dimensi yaitu  $(x, y)$  atau  $(x, y, z)$ . Untuk hipotesis ke- $i$  pada tangkapan ke- $k$ , dinotasikan  $\Omega_i^k$ . Setiap  $\Omega_i^k$  berisi target yang telah terbentuk sebelumnya  $T_i^k$  ( $i \in 1... n \text{ target}$ ), *target* estimasi, estimasi kovarian dan asosiasi  $\psi_i^k$ , antara tangkapan  $Z^k$  dan target hipotesis  $T_i^k$ . Setiap hipotesis  $\Omega_i^k$  diasosiasikan dengan  $p_i^k$  yaitu probabilitas dari setiap hipotesisnya.

Setiap  $k$  pada hipotesis  $\Omega^{k-1}$  digunakan untuk menghasilkan hipotesis dari  $\Omega^k$  yaitu sebuah hipotesis pada waktu ke- $k$ . Setiap hipotesis  $\Omega_j^{k-1}$  merupakan sebuah hipotesis baru yang dibentuk  $j\Omega^k$  yang mempunyai  $\Omega_j^{k-1}$  merupakan hipotesis *parent*. Pada setiap generasi baru hipotesis  $j\Omega^k$  untuk setiap  $Z_l^k$  berisi tiga macam variabel yaitu *false alarm* (FA), target baru dan sebuah target yang sudah terbentuk sebelumnya. Hasil tangkapan  $Z_l^k$  akan menjadi suatu target  $T_i^k$  pada hipotesis  $\Omega_j^{k-1}$  jika memenuhi validasi yaitu berada di dalam wilayah posisi yang diharapkan. Sebuah target hanya akan terikat dengan satu hasil tangkapan dan tidak mungkin target akan terikat dengan lebih dari satu tangkapan. Target yang tidak mengalami penambahan hasil tangkapan selama  $n$  tangkapan akan dihapus.



Gambar 5. Alur Kerja Algoritma MHT

Probabilitas dari hipotesis baru  $j\Omega_i^k$  yang diberikan dari hipotesis yang sebelumnya  $\Omega_j^{k-1}$  dan hasil tangkapan  $Z^k$  [17], adalah seperti pada Persamaan 1.

$$j p_i^k = \frac{1}{c} x P_D^{N_d} x (1 - P_d)^{N_t - N_d} x (P_{FA})^{N_{fa}} x (P_{NT})^{N_{nt}} x \prod_{(z_i^k, \iota_{T_i}^k) \in \psi_i^k} P_{z_i^k, \iota_{T_i}^k} x p_i^{k-1} \quad (1)$$

Dimana  $N_d$  menyatakan nilai dari hasil tangkapan atau *measurement* dan  $N_t$  merupakan jumlah dari *target* pada  $\Omega_j^{k-1}$ ,  $N_{fa}$  adalah jumlah *false alarm* dan  $N_{nt}$  adalah jumlah *target* baru. Selanjutnya  $P_d$  adalah nilai probabilitas deteksi *target*,  $P_{FA}$  merupakan probabilitas tangkapan *false alarm* dan  $P_{NT}$  adalah probabilitas deteksi *target* baru. Probabilitas hipotesis *parent*  $p_j^{k-1}$  dan  $P_{z_i^k, \iota_{T_i}^k}$  merupakan probabilitas *measurement*  $Z_i^k$  yang dideteksi pada *target*  $\iota_{T_i}^k$ .

Algoritma ini bisa menimbulkan pembentukan kombinasi yang sangat banyak. Untuk mengatasi hal itu, digunakan proses *pruning* dari pohon hipotesis. Proses *pruning* tersebut dilakukan dengan cara pembatasan nilai daun atau kedalaman dari pohon hipotesis tersebut [18]. Tetapi kadang-kadang juga ditemui kejadian bahwa hipotesis ketika mengalami *pruning* terjadi proses yang bisa memakan waktu dan memori yang terlalu tinggi, sehingga kejadian seperti itu tidak akan efektif. Maka sebuah solusi terciptakan dengan menggunakan algoritma Murty untuk menemukan *m-best* penugasan untuk asosiasi setiap daun [7].

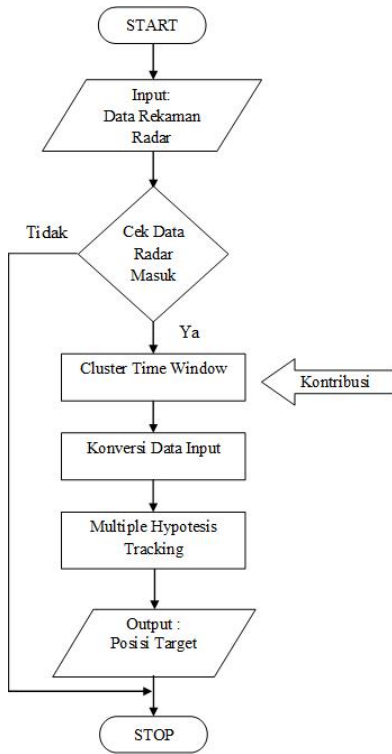
Metode ini cukup baik digunakan untuk kasus MTT [5]. MHT yang digunakan berdasarkan algoritma Reid (Blackman, 2004) untuk meningkatkan efisiensi, dikombinasikan dengan algoritma Murty [7]. Alur sistem algoritma MHT terlihat pada Gambar 5.

Berdasarkan Gambar 5 dapat diperhatikan bahwa input data yang berupa empat macam variabel parameter yaitu  $x$ ,  $y$ , *heading* dan *altitude* yang dihasilkan pada proses konversi data akan diproses di dalam algoritma MHT. Pertama memasuki proses *Gating*, dimana *target* akan mengalami validasi data berdasarkan data *track* sebelumnya. Biasanya akan berbentuk wilayah atau *region*, dimana *target* sementara yang masuk pada wilayah tersebut akan menjadi calon *track*. Setelah mengalami proses *Gating*, *target* yang lolos proses validasi akan mengalami proses pembentukan hipotesis dan akan dipilih hipotesis terbaik yang akan dibentuk menjadi *track* baru dengan menggunakan algoritma Murty dengan *library* yang dikembangkan oleh Orguner [19]. Untuk evaluasinya, *track* atau hipotesis yang tidak mengalami penambahan anggota dari *track* yang bersangkutan selama  $n$  kali *update* maka *track* atau hipotesis tersebut akan dihapus. Proses prediksi *track* akan menggunakan Kalman Filter standar yang akan menggunakan *library* [16]. Proses ini berfungsi untuk menentukan estimasi atau prediksi posisi dari *target* yang diinginkan kemudian dari *target* hasil estimasi tersebut digunakan untuk melakukan perhitungan probabilitas ketika proses pembentukan hipotesis. *Target* estimasi ini digunakan untuk menghitung nilai probabilitas dengan melibatkan data calon *target* hasil tangkapan yang baru masuk dengan *target* estimasi hasil prediksi dari Kalman Filter dengan menggunakan *target* yang sudah terbentuk dari proses *tracking* sebelumnya. Hasil nilai probabilitas tersebut akan digunakan sebagai bobot untuk menentukan hipotesis terbaik dengan menggunakan algoritma Murty pada implementasi untuk algoritma MHT menggunakan *library* yang dibuat oleh Antunes [20].

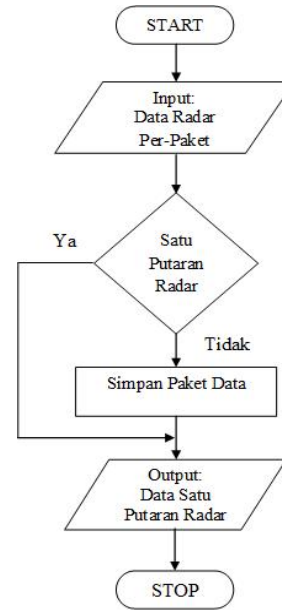
### G. Perancangan Algoritma Sistem

Perancangan algoritma merupakan proses yang sangat penting untuk mendapatkan hasil yang diinginkan. Maka dengan pertimbangan tersebut, penyusunan langkah secara detil dan teliti sangat diperlukan. Dalam penyusunan setiap prosesnya, juga perlu memperhatikan hasil setiap outputnya sehingga perencanaan untuk setiap langkahnya perlu dilakukan uji secara manual. Sistem MTT memiliki langkah-langkah utama yang harus dijalankan, yaitu *track association*, *track smoothing*, *track initiation* dan *track maintenance*.





Gambar 6. Alur Sistem MTT dengan *Clustering Time Window*



Gambar 7. Alur Sistem *Clustering Time Window*

Berdasarkan Gambar 6, sistem ini nantinya terdiri dari tiga tahapan utama yaitu untuk tahap pertama merupakan *clustering time window*. Pada tahap ini, data rekaman radar akan dikluster berdasarkan waktu satu kali putaran radar pada data rekaman yang telah direkam langsung dari radar yang telah disiapkan. Tahap kedua adalah konversi data radar sebelum digunakan oleh algoritma MHT, pada tahap ini data radar masih berupa nilai *range* dan *bearing* untuk memudahkan proses di dalam algoritma MHT maka akan dikonversi ke dalam operasi kartesian yaitu berbentuk nilai *x* dan *y*. Untuk tahap yang terakhir yaitu data yang sudah dikonversi tersebut akan digunakan untuk input algoritma MHT. Alur tersebut terus berulang sampai data rekaman yang telah disiapkan habis untuk diproses. Berdasarkan hasil MHT yang berupa beberapa posisi *target* yang telah membentuk *track* atau lintasan tersebut akan dievaluasi dengan data *ground truth* yang telah disediakan. Penyiapan data *ground truth* ini adalah dengan menggunakan data rekaman yang sama tetapi proses pencocokan *track* atau lintasan masing-masing *target* dengan menggunakan mode 3 yang merupakan identitas dari masing-masing pesawat. Untuk mengetahui kemampuan algoritma dilakukan dengan membandingkan antara *track* atau lintasan yang dibentuk dengan algoritma yang diusulkan dengan lintasan yang dibentuk berdasarkan algoritma MHT murni tanpa adanya proses *clustering time window*.

Berdasarkan Gambar 7, tentang alur sistem CTW masukan data radar perpaket data maksudnya data akan diproses setiap satu paket *target* (pesawat) yang terdiri variabel waktu, *range*, *bearing*, mode 3 dan *altitude*. Aliran paket tersebut akan disimpan selama satu putaran radar penuh dan setelah satu putaran radar selesai maka beberapa paket data yang tersimpan akan digunakan untuk input pada proses konversi data input.

Konversi data input ini memiliki tujuan untuk mempermudah penghitungan di dalam metode MHT. Data yang pada awalnya tadi berupa nilai *range* (*r*) dan *bearing* ( $\theta$ ) dikonversi terlebih dahulu dengan operator kartesian *x* dan *y*. Rumus yang diberikan untuk konversi tersebut terdapat pada Persamaan 2 dan 3.

$$x = r \cos \theta \tag{2}$$

$$y = r \sin \theta \tag{3}$$

Pada proses ini, juga dilakukan perhitungan kecepatan (*speed*) dan *heading*. *Heading* pesawat adalah sudut yang dibentuk antara arah hidung pesawat dengan *true north*. Berikut ini adalah persamaan untuk menghitung nilai kecepatan dan *heading* dari setiap pesawat.

$$dx = (x_t - x_{t-1}) \tag{4}$$

$$dy = (y_t - y_{t-1}) \tag{5}$$

$$jarak = \sqrt{dx^2 + dy^2} \tag{6}$$

$$kecepatan = \frac{jarak}{waktu_t - waktu_{t-1}} \quad (7)$$

Berdasarkan pada persamaan-persamaan di atas,  $dx$  adalah selisih antara nilai  $x$  pada waktu ke  $t$  dengan nilai  $x$  pada waktu sebelumnya  $t-1$ . Kemudian  $dy$  adalah selisih nilai  $y$  pada waktu ke  $t$  dengan nilai  $y$  pada waktu sebelumnya  $t-1$ . Dari akar kuadrat nilai  $dx$  dan  $dy$  didapatkan jarak antara kedua titik  $(x_t, y_t)$  dengan  $(x_{t-1}, y_{t-1})$ . Selanjutnya kecepatan (*speed*) dapat diperoleh berdasarkan perbandingan nilai jarak dengan selisih dari waktu pada saat  $t$  yaitu  $waktu_t$  dengan waktu pada saat  $t-1$  yaitu  $waktu_{t-1}$ . Sedangkan untuk memperoleh nilai dari *heading* masing-masing target, yaitu menggunakan persamaan sebagai berikut.

$$\alpha = \tan^{-1}\left(\frac{dy}{dx}\right) - 90 \quad (8)$$

$$heading = \begin{cases} -\alpha, & \text{untuk } \alpha < 0 \\ 360 - \alpha, & \text{untuk } \alpha \geq 0 \end{cases} \quad (9)$$

Berdasarkan persamaan 7 dan 8, nilai *heading* diperoleh dengan menghitung  $\tan^{-1}\left(\frac{dy}{dx}\right)$  yang dikurangi dengan 90, kemudian didapat nilai dari  $\alpha$ . Selanjutnya nilai *heading* akan didapatkan minus  $\alpha$  jika nilai  $\alpha$  kurang dari 0. Sebaliknya, jika nilai  $\alpha$  lebih dari 0 nilai *heading* adalah selisih 360 dengan  $\alpha$ . Sehingga dari beberapa variabel yang terdiri  $x$ ,  $y$ , *speed*, *altitude* dan *heading*, akan menjadi atribut untuk menjadi sebuah data *target* yang akan menjadi input untuk algoritma MHT.

#### IV. SKENARIO UJI COBA

Pada penelitian ini penulis menggunakan beberapa perangkat keras dan perangkat lunak untuk menunjang proses implementasinya. Perangkat keras yang digunakan untuk implementasi dan pengujian adalah sebuah laptop dengan spesifikasi processor Intel Core i5-3210M CPU@2.50 GHz, RAM 4 Gb. Selain perangkat keras, juga menggunakan perangkat lunak untuk implementasi dan pengujiannya yaitu peneliti menggunakan sistem operasi Windows 7 32 bit, NetBeans IDE 8.0.1 dan Matlab R2013a.

Data yang digunakan dalam penelitian ini adalah data rekaman berdurasi  $\pm 18$  menit. Data tersebut adalah milik salah satu perusahaan yang bergerak dalam bidang pertahanan militer. Data yang digunakan ini berupa data yang terdiri *time arrival*, *range*, *bearing*, *mode3* dan *altitude*. Mode 3 adalah identitas sebuah pesawat yang dengan menggunakan identitas tersebut setiap plot atau titik akan bisa dihubungkan menjadi sebuah lintasan atau *track* yang bisa digunakan sebagai lintasan aktual (*ground truth*).

##### A. Uji Coba Pengelompokan Data Radar Menggunakan *Clustering Time Window*

Metode ini merupakan metode yang peneliti ajukan dalam penelitian ini. CTW merupakan suatu metode pengelompokan data berdasarkan satu putaran radar, dimana alur sistem kerjanya seperti pada Gambar 7. Dalam melakukan satu putaran radar tersebut diperlukan waktu  $\pm 10$  detik. CTW ini bertujuan untuk memastikan bahwa satu kelompok data radar yang akan digunakan untuk proses *tracking* tidak mengalami *double target* atau dua pesawat yang sama. Misalkan pada contoh data pada TABEL II, untuk data dengan Mode3 902 dalam kelompok satu proses CTW tersebut tidak ada data Mode3 902 yang lain. Hal ini sangat berfungsi ketika digunakan untuk proses *tracking* terutama ketika menggunakan metode pengembangan MHT karena dalam metode MHT tersebut dalam satu proses *tracking* hanya bisa menambah plot masing-masing *track*, hanya satu buah plot dalam satu *state* ketika melakukan *update*. Dengan kondisi seperti itu, bisa dianalisa bahwa ketika suatu proses pengelompokan data tersebut berisi Mode3 dalam hal ini 902 dengan jumlah lebih dari satu dalam kelompok data tersebut, maka bisa dipastikan satu plot Mode3 902 yang lain bisa membentuk *track* atau lintasan baru atau bisa juga ada kemungkinan plot tersebut akan bergabung dengan *track* yang lain. Jika sebuah *track* mengalami kondisi tersebut, maka lintasan dari pesawat tersebut menjadi tidak valid, karena pasti lintasan dari pesawat tersebut bisa putus sehingga ada informasi dari *track* tersebut yang akan hilang. Pada TABEL II merupakan salah satu contoh hasil *scan* atau tangkapan CTW.

TABEL II  
PENGELOMPOKAN DATA RADAR DENGAN CTW PADA SCAN PERTAMA

Range	Bearing	Mode3	Altitude
46	73.84615	1332	331
90	248.6154	903	314
21	235.2527	908	191
98	277.0989	0	2
163	54.85714	1694	2
117	220.8352	902	220
144	61.18681	3673	388



56	212.3956	906	242
----	----------	-----	-----

TABEL III  
 PENGELOMPOKAN DATA RADAR TANPA CTW PADA SCAN PERTAMA

Range	Bearing	Mode3	Altitude
46	73.84615	1332	331
90	248.6154	903	314
118	220.8352	902	220
162	54.85714	1694	400
21	235.2527	908	191
98	277.0989	0	2
163	54.85714	1694	2
0	0	0	0
117	220.8352	902	220
144	61.18681	3673	388
56	212.3956	906	242

Dari tangkapan data radar tersebut, bisa diamati jika setiap tangkapan satu putaran radar sudah berhasil tidak ada data dengan Mode3 yang lebih dari satu dalam setiap *scan* CTW. Hal tersebut akan sangat membantu dalam proses *tracking* dengan metode MHT yang digunakan, karena bisa mengurangi kesalahan proses pelacakan.

**B. Uji Coba Pengelompokan Data Radar Tanpa Menggunakan *Clustering Time Window***

Metode ini adalah metode sebelumnya dimana pengelompokan data tidak menggunakan satu putaran radar atau hanya berdasarkan durasi waktu tertentu. Pada proses ini, data dikelompokkan secara tidak teratur, jadi kalau bisa diamati data bisa saja tepat satu putaran atau bahkan tidak tepat satu putaran sama sekali. Pada proses pengelompokan tanpa CTW ini, data yang masuk dikelompokkan dalam kurun waktu yang tidak tentu sehingga pada hasil pengelompokan datanya sering tertangkap data dengan pesawat yang lebih dari satu dalam kelompoknya. Hal tersebut bisa terlihat pada plot-plot yang tertangkap pada masing-masing *scan* atau hasil tangkapan. Pada implementasi metode ini, memang lebih banyak hasil pengelompokan yang setiap kelompok data radarnya berisi Mode3 yang sama atau lebih dari satu. Pada TABEL III merupakan salah satu contoh hasil *scan* atau tangkapan tanpa CTW.

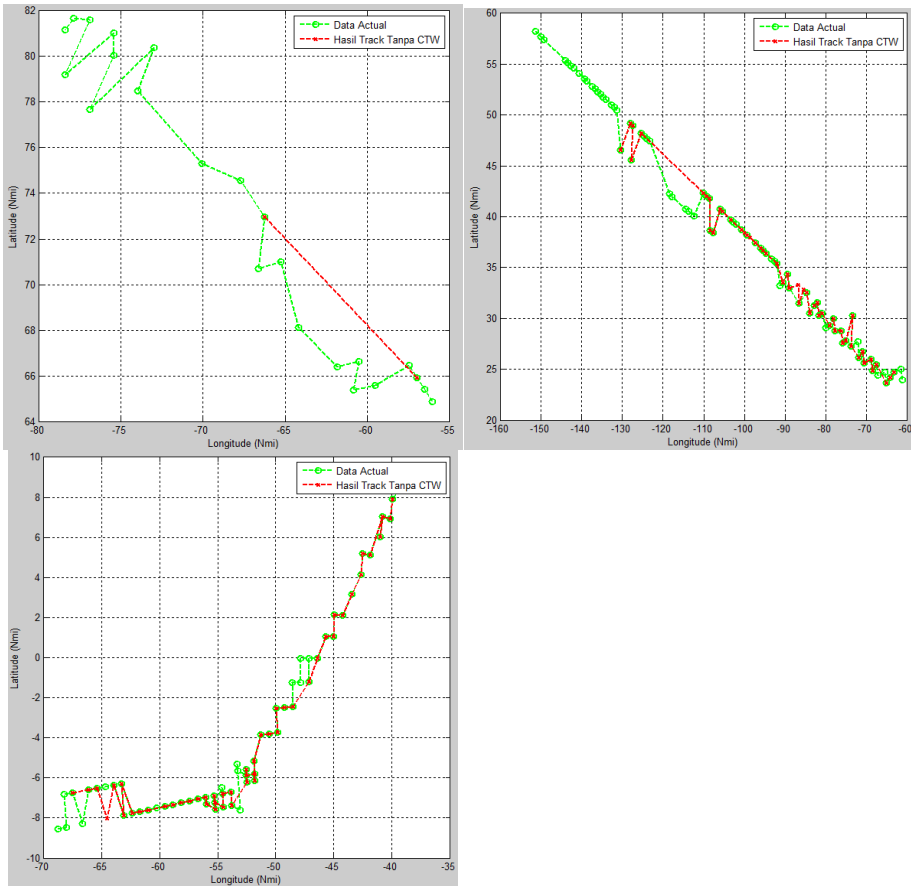
Dari sini, bisa dianalisa ketika menggunakan algoritma MHT dengan anggota kelompok dalam setiap *scan* atau tangkapannya memiliki plot dengan Mode3 yang tidak unik atau lebih dari satu, kemungkinan untuk salah *tracking* akan semakin besar. Nilai *range*, *bearing*, *mode3* dan *altitude* yang semuanya bernilai 0 merupakan salah satu bentuk data *noise* yang terjadi ketika proses pengelompokan data radar tanpa CTW. Kasus tersebut terjadi ketika proses pengelompokan dilakukan tetapi data yang diproses belum dapat diterima sehingga diperoleh nilai 0 semua pada data ke-8 tersebut. Karena pada proses pengelompokan data radar tanpa CTW proses dilakukan secara terus-menerus tanpa melihat data tersebut sudah masuk atau belum, dimana proses ini hanya melihat berdasarkan lama waktu tertentu saja.

**C. Uji Coba Performa Tracking Tanpa Clustering Time Window dengan Menggunakan Pengembangan Multiple Hypothesis Tracking**

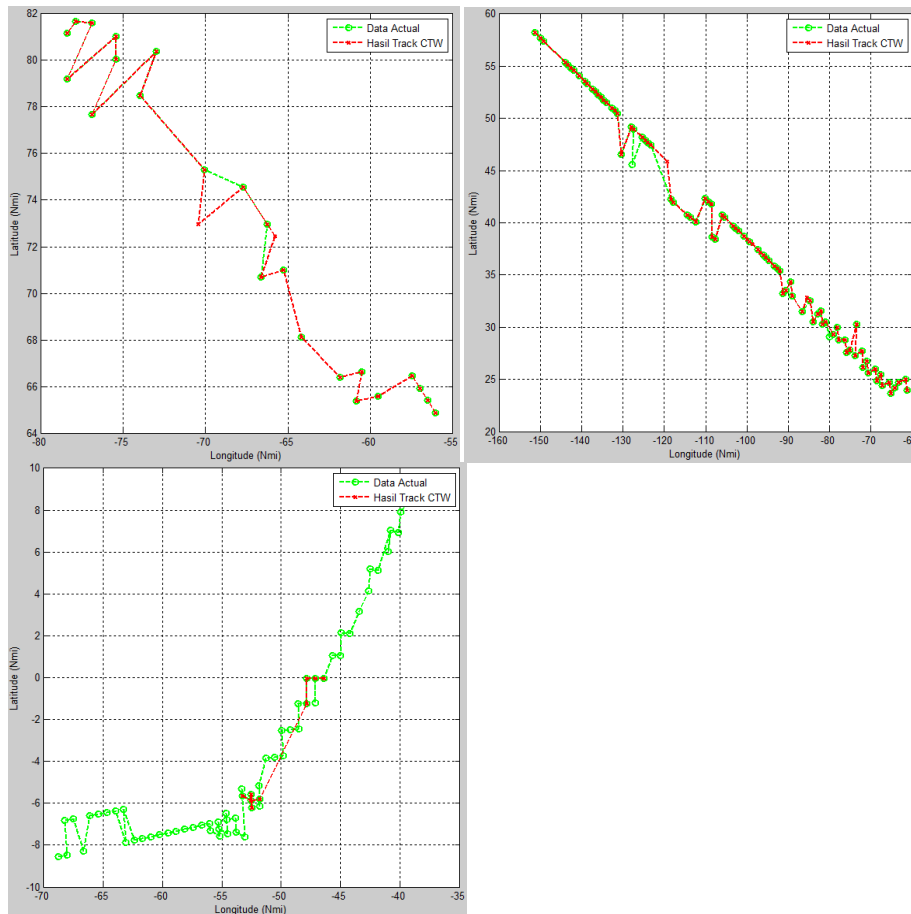
Berdasarkan Gambar 8, terlihat bahwa hasil *tracking* untuk pesawat dengan Mode3 902 sangat jelek karena hanya bisa melakukan *tracking* sebanyak dua plot saja pada data aktual (*ground truth*), kemudian untuk data pesawat pada Mode3 903 sudah lumayan banyak plot atau titik yang berhasil di-*track* berdasarkan data aktualnya. Dan yang terakhir adalah data pesawat pada Mode3 958. Pada data ini, hasil *tracking* sudah lebih bagus dibandingkan dua data pesawat sebelumnya. Hal itu bisa dilihat dari banyaknya plot pada data aktual yang berhasil di-*track*. Pada kasus MHT tanpa CTW ini, penyebab banyak *track* atau lintasan setiap pesawat tidak berhasil adalah karena pada metode MHT tidak mengenal sistem penambahan atau *update track* sebanyak lebih dari satu plot atau titik, tetapi hanya bisa menambah satu plot saja dalam sekali proses *tracking* dilakukan. Padahal pada kondisi MHT tanpa CTW ini, bisa dimungkinkan setiap sekali proses bisa terjadi dua kali *update*.

**D. Uji Coba Performa Tracking Menggunakan Clustering Time Window dengan Pengembangan Multiple Hypothesis Tracking**

Pada bagian ini akan dilakukan uji coba untuk metode yang diusulkan yaitu MHT-CTW, dimana hasilnya menunjukkan *tracking* yang cukup bagus. Seperti ditampilkan pada Gambar 9, dari tiga lintasan pesawat yang ditampilkan dua *track* menunjukkan hasil yang sempurna sedangkan satu *track* mendapatkan hasil yang kurang bagus.



Gambar 8. Dari Kiri ke Kanan : Hasil *Tracking* Pesawat dengan Mode3 902, Mode3 903, Mode3 958



Gambar 9. Dari Kiri ke Kanan: Hasil *Tracking* Pesawat dengan Mode3 902, Mode3 903, Mode3 958

TABEL IV  
URUTAN KEDATANGAN DATA REKAMAN MEMILIKI NOISE

Mode3	X(Lon)	Y(Lat)	Heading	Altitude
1694	95.2059	-67.018	0	2
902	-54.646	63.2296	0	220
903	-59.86	23.4403	0	314
908	-12.325	8.54936	0	191
0	-69.463	-8.6508	0	2
3673	90.123	-49.573	0	388
1332	31.5599	-9.1414	0	331
906	-21.43	33.7748	0	242
903	-60.23	24.4419	0	319
958	-69.463	-8.6508	0	2
1694	94.6218	-66.607	0	400
908	-12.857	9.03555	0	192

Gambar 9 merupakan hasil *tracking* dengan metode yang diajukan yaitu MHT-CTW. Seperti dapat diamati bahwa untuk data pesawat dengan Mode3 902 dan 903 berhasil melakukan proses *tracking* dengan hampir sempurna karena semua data aktualnya berhasil di *track*. Sedangkan untuk kasus Mode3 958 adalah yang terburuk karena disebabkan oleh plot *noise* di awal data kedatangan. Salah satu kasus dimana hasil dari MHT-CTW kurang memperlihatkan hasil yang kurang bagus terjadi pada Mode3 958.

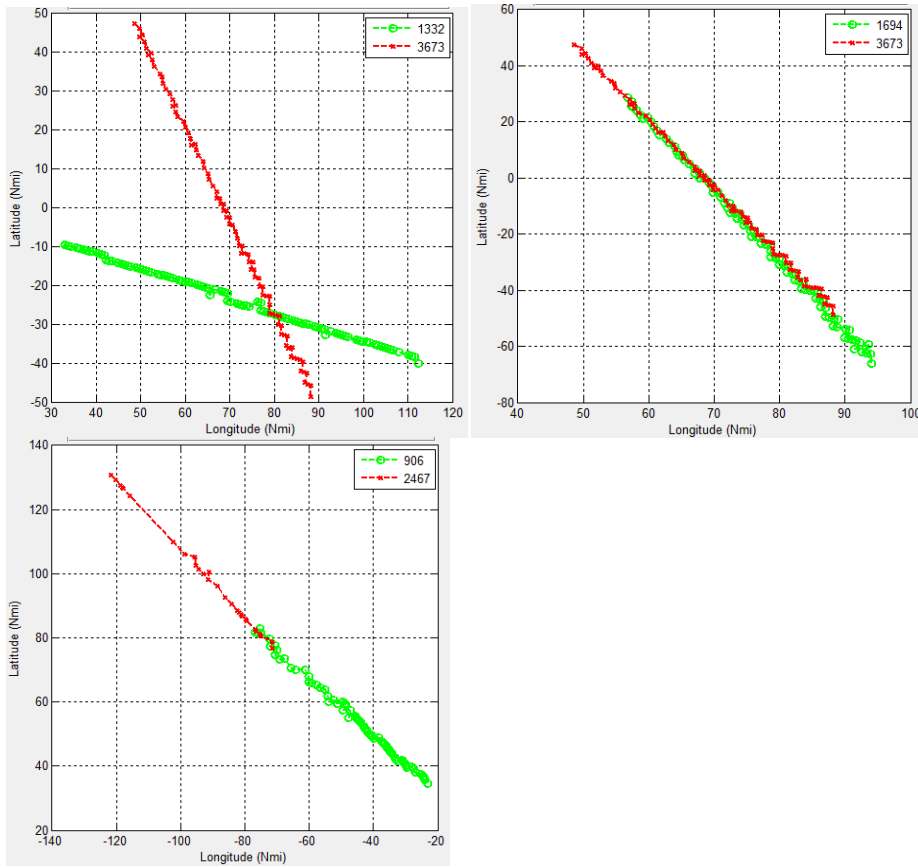
Berdasarkan Gambar 9 di atas, dapat terlihat MHT-CTW masih kalah dominan daripada MHT tanpa CTW. Kasus ini juga menjadi salah satu kekurangan pada MHT-CTW, khususnya dalam menghadapi data yang memiliki *noise* seperti pada kasus Mode3 958 ini. *Noise* tersebut terjadi ketika suatu data dengan Mode3 0 seperti pada TABEL IV terjadi yaitu ketika data tersebut memiliki data yang sama dengan Mode3 958. Dengan kondisi bahwa Mode3 0 datang lebih dahulu, secara otomatis data tersebut tersimpan sebagai *track* dengan identitas tertentu, kemudian ketika Mode3 958 datang berikutnya dengan kondisi data yang sama maka secara tidak langsung plot dengan Mode3 958 tersebut akan tersambung dengan *track* yang sudah terbentuk sebelumnya dalam hal ini yaitu plot dengan Mode3 0 tadi. Selanjutnya secara otomatis Mode3 958 tersambung dengan lintasan dengan Mode3 0 tersebut yang berakibat lintasan pada Mode3 958 sendiri tidak berhasil melakukan proses *tracking* dengan sempurna.

Berdasarkan kasus tersebut bisa diambil kesimpulan bahwa metode yang diusulkan ini memiliki salah satu kelemahan yaitu ketika pembentukan *track* atau lintasan salah pada waktu kedatangan di awal maka untuk proses pembentukan lintasan selanjutnya akan ikut mengalami kesalahan. Hal ini bisa terjadi pada salah satu kasus pada Mode3 958 ini. Sehingga lintasan pada Mode3 958 tidak begitu *valid* karena banyaknya plot dari lintasan tersebut yang hilang jika dilihat berdasarkan data aktual yang dimiliki oleh data pada Mode3 958 ini.

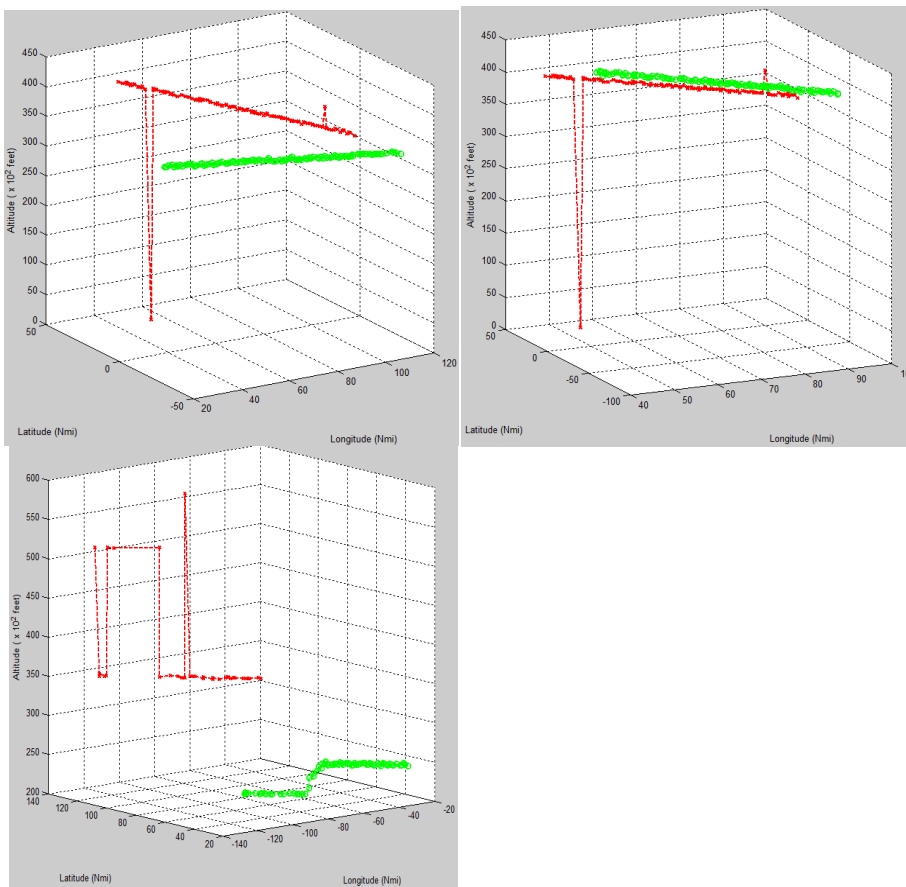
#### E. Uji Coba Dua *Target* Dalam Kondisi Komplek

Pada bagian ini, akan ditampilkan hasil uji coba untuk tiga macam kasus lintasan yang cukup komplek, yaitu lintasan saling bersilangan, lintasan saling beriringan dan lintasan saling berhadapan. Pada kondisi-kondisi seperti itu, sangat bisa dimungkinkan akan terjadi kesalahan *tracking* karena memang kondisinya yang cukup membingungkan jika dilihat secara sepintas.

Berdasarkan Gambar 10, ternyata metode MHT-CTW juga cukup handal ketika melakukan proses *tracking* pada ketiga macam kondisi yang sebenarnya cukup sulit untuk dilakukan proses *tracking*. Ketiga kondisi tersebut adalah lintasan saling bersilangan, lintasan saling beriringan dan lintasan saling berhadapan. Karena ketiga kondisi tersebut sangat memungkinkan bisa menyebabkan kesalahan *tracking* yang cukup tinggi. Untuk lebih jelas menggambarkan sebenarnya lintasan tersebut saling tabrakan atau tidak, bisa dilihat berdasarkan tampilan 3D pada Gambar 11.



Gambar 10. Hasil Tracking 2D MHT-CTW dari Kiri ke Kanan: *Track Saling Bersilangan, Track Saling Beriringan, Track Saling Berhadapan*



Gambar 11. Hasil Tracking 3D MHT-CTW dari Kiri ke Kanan: *Track Saling Bersilangan, Track Saling Beriringan, Track Saling Berhadapan*

TABEL V  
PERBANDINGAN RATA-RATA NILAI CT, UT, MT, IT SELURUH HASIL SCAN  
PADA MHT TANPA CTW DAN MHT-CTW

Metode	CT (%)	UT (%)	MT (%)	IT (%)
MHT Tanpa CTW	51.49	46.38	29.73	46.73
MHT -CTW	87.66	12.81	80.5	23.65

Berdasarkan Gambar 11 tersebut, sudah jelas bagaimana sebenarnya posisi dari masing-masing pesawat tersebut, yaitu kondisinya cukup aman dan tidak akan terjadi tabrakan atau *crash* di udara. Bisa diamati dari selisih *altitude* atau ketinggian dari kedua *target* tersebut.

### V. HASIL DAN PEMBAHASAN

Pada bagian hasil dan pembahasan akan diamati performa proses *tracking* baik MHT tanpa CTW dan MHT-CTW berdasarkan tiga nilai yaitu *Correct Target* (CT), *Undetected Target* (UT), *Maintain Target* (MT) dan *Inexisting Target* (IT). CT merupakan variabel nilai untuk mengetahui seberapa besar kemampuan sistem untuk mendeteksi suatu plot yang diprediksi sebagai *target* kemudian pada proses *output tracking* berhasil benar-benar menjadi *target* dari salah satu pesawat. UT merupakan variabel nilai yang menyatakan sebuah target tetapi setelah proses *tracking* tidak terdeteksi sebagai target. MT merupakan variabel nilai untuk mengetahui seberapa besar kemampuan sistem untuk mendeteksi suatu plot yang baru masuk atau tertangkap, berhasil tersambung dengan plot sebelumnya yang sudah menjadi *track* pada salah satu *target* atau secara sederhana bisa dikatakan untuk menghitung berapa jumlah plot yang berhasil tersambung dengan *track* pasangannya. IT merupakan variabel nilai untuk mengetahui seberapa besar kemampuan sistem untuk mendeteksi suatu plot dimana diprediksi akan menjadi sebuah *target* tetapi pada kenyataan pada hasil *output tracking* tidak menjadi sebuah *target*. Persamaan untuk mendapatkan keempat nilai tersebut adalah sebagai berikut.

$$CT = \frac{\text{jumlah correct target}}{(\text{jumlah correct target} + \text{jumlah undetected target})} \times 100\% \tag{10}$$

$$UT = \frac{\text{jumlah undetected target}}{(\text{jumlah correct target} + \text{jumlah undetected target})} \times 100\% \tag{11}$$

$$MT = \frac{\text{jumlah maintain target}}{\text{jumlah correct target}} \times 100\% \tag{12}$$

$$IT = \max \left\{ 0, \min \left\{ 1, \frac{\text{jumlah inexisting target}}{(\text{jumlah correct target} + \text{jumlah undetected target})} \right\} \right\} \times 100\% \tag{13}$$

Berdasarkan pada TABEL V, terlihat perbandingan metode terdahulu dengan metode MHT-CTW dengan menggunakan seluruh data rekaman yang berdurasi kurang lebih 18 menit. Dari hasil tersebut terlihat perbedaan yang cukup mencolok. Untuk nilai CT, metode terdahulu dengan nilai 51.49% sedangkan metode MHT-CTW memiliki nilai 87.66%. Hal itu menunjukkan bahwa kemampuan mendeteksi target untuk metode MHT-CTW lebih bagus. Kemudian pada nilai UT metode terdahulu dengan nilai 46.38% sedangkan metode MHT-CTW memiliki nilai 12.81%. Dari hasil tersebut berarti cukup baik menggunakan metode MHT-CTW karena target yang tidak terdeteksi atau hilang cukup kecil. Sedangkan pada nilai MT pada metode terdahulu menunjukkan nilai 29.73% sedangkan metode MHT-CTW memiliki nilai 80.5%. Dari sini bisa juga dikatakan bahwa metode MHT-CTW memiliki kemampuan yang cukup bagus dalam hal membentuk lintasan setiap pesawat khususnya dalam menghubungkan setiap plot dengan tepat pada setiap lintasan pesawat tersebut. Dan yang terakhir, untuk nilai IT pada metode terdahulu cukup besar dibandingkan dengan metode MHT-CTW yaitu 46.73% dibandingkan dengan 23.65%. Hal itu mengandung maksud bahwa metode terdahulu memiliki kecenderungan memiliki tingkat kesalahan prediksi yang lebih tinggi dibandingkan dengan metode MHT-CTW. Dari beberapa hasil analisa tersebut dapat ditarik kesimpulan awal bahwa dengan penambahan praproses berupa metode CTW pada MHT menghasilkan sistem *tracking* yang cukup handal.

### VI. KESIMPULAN

Pada penelitian ini dapat disimpulkan bahwa Sistem *multitarget tracking* dengan *Multiple Hypothesis Tracking* dengan praproses *Clustering Time Window* pada data radar bisa mendapatkan hasil yang cukup bagus dengan klasifikasi nilai *correct target* sebesar 87.66%, *undetected target* sebesar 12.81%, *maintain target* sebesar 80.5% dan *inexisting target* sebesar 23.65%. Dan juga dapat diamati bahwa Sistem *multitarget tracking* dengan MHT-CTW juga unggul dalam beberapa percobaan data dengan memakai data aktual (*ground truth*) sebagai acuannya, yang ditampilkan untuk membentuk suatu *track* dari setiap pesawat dibandingkan dengan hasil *tracking* dari metode MHT tanpa CTW. Selain hasil penelitian yang bisa dibilang cukup bagus, di dalam metode yang telah diajukan juga ditemukan kelemahan yaitu ketika ada penangkapan data *noise* di awal penangkapan data. Pada

metode ini, ketika ada salah *tracking* satu plot atau titik di awal maka untuk *track* selanjutnya juga ikut salah sehingga diperlukan studi lebih lanjut untuk mengatasi masalah tersebut.

## DAFTAR PUSTAKA

- [1] Pineda, N. A. (2013, July 2). Radar Tracking System Using Contextual Information on a Neural Network Architecture in Air Combat Maneuvering. pp. 1-11.
- [2] Strauss, C., & Stephany, S. (2011). Spatio-temporal clustering with a temporal sliding window for tracking of convective activity. *XI Workshop de Computação Aplicada*. INPE.
- [3] Chunki Park, H.-T. L. (2011). Radar Data Tracking Using Minimum Spanning Tree-Based Clustering Algorithm. *11th AIAA Aviation Technology, Integration, and Operations (ATIO) Conference* (pp. 1-8). Moffet Field, USA: AIAA.
- [4] Xue Jiang, K. H. (2013, July 4). Integrated track initialization and maintenance in heavy clutter using probabilistic data association. *Signal Processing*, pp. 241-250.
- [5] Blackman, S. S. (2004). Multiple Hypothesis Tracking For Multiple Target Tracking. *SYSTEMS MAGAZINE*, 5-17.
- [6] Wolf, M. T., & Burdick, J. W. (2009). Multiple Hypothesis Tracking Using Clustered Measurement. *International Conference on Robotics and Automation* (pp. 3955-3961). Kobe: IEEE.
- [7] Cox, I. J., & Hingorani, S. L. (1996). An Efficient Implementation of Reid's Multiple hypothesis Tracking Algorithm and Its Evaluation for the Purpose of Visual Tracking. *TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE*, 138-150.
- [8] Fortunato, E., Kreamer, W., Mori, S., Chong, C.-Y., & Castanon, G. (2007). Generalized Murty's Algorithm With Application to Multiple Hypothesis Tracking. *Information Fusion*, 1-8.
- [9] Makris, A., & Prieur, C. (2014). Bayesian Multiple-Hypothesis Tracking of Merging and Splitting Targets. *Geoscience and Remote Sensing*, 7684 - 7694.
- [10] Richards, M. A., Scheer, J. A., & Holm, W. A. (2010). *Principle Of Modern Radar*. Danvers: Sheridan Books, Inc.
- [11] Chandra, F. (2007). *Algoritma Filtering Pada Sistem Multi Radar Tracking*. Bandung: Institut Teknologi Bandung.
- [12] Murty, K. G. (1968). An Algorithm for Ranking all the Assignments in Order of Increasing Cost. *Operations Research*, 682-687.
- [13] Singer, R., & Stein, J. (1971). An optimal tracking filter for processing sensor data of imprecisely determined origin in surveillance systems. *Decision and Control, 1971 IEEE Conference on* (pp. 171-175). Miami Beach: IEEE.
- [14] Reid, D. B. (1979). An Algorithm for Tracking Multiple Targets. *TRANSACTIONS ON AUTOMATIC CONTROL*, 843-854.
- [15] Amditis, A., Thomaidis, G., Maroudis, P., Lytrivis, P., & Karaseitanidis, G. *Multiple Hypothesis Tracking Implementation*. Greece: www.intechopen.com.
- [16] Abdelkader, A. (2009, December 6). *The Lost Beauty*. Retrieved September 18, 2014, from Java implementation of the Kalman Filter using JAMA: <http://the-lost-beauty.blogspot.com/2009/12/java-implementation-of-kalman-filter.html>
- [17] Antunes, D. M., Figueira, D., Matos, D. M., Bernardino, A., & Gaspar, J. (2011). Multiple Hypothesis Tracking in Camera Networks. *International Conference on Computer Vision Workshops* (pp. 367-374). Barcelona: IEEE.
- [18] Bar-Shalom, Y., Daum, F., & Huang, J. (2009). The Probabilistic Data Association Filter. 82-100.
- [19] Orguner, U. (2010, January 26). *Linkopings Universitet*. Retrieved September 25, 2014, from Reglerteknik (Automatic Control): <http://www.control.isy.liu.se/student/graduate/TargetTracking/Murty.m>
- [20] Antunes, D. (2011, May 10). *Multiple Hypothesis Library*. Retrieved September 4, 2014, from Multiple Hypothesis Library Easy MHT Implementation: <http://www.multiplehypothesis.com>