

Christer Wigren

Realtidssimuleringar av målsökare och motmedel (rök, vattendimma) i en komplex bakgrundsmiljö



TOTALFÖRSVARETS FORSKNINGSINSTITUT

Ledningssystem
Box 1165
581 11 Linköping

FOI-R--0694--SE

december 2002

ISSN 1650-1942

Metodrapport

Christer Wigren

Realtidssimuleringar av målsökare och motmedel (rök, vattendimma) i en komplex bakgrundsmiljö

Utgivare Totalförsvarets Forskningsinstitut - FOI Ledningssystem Box 1165 581 11 Linköping	Rapportnummer, ISRN FOI-R--0694--SE	Klassificering Metodrapport
	Forskningsområde 6. Telekrig	
	Månad, år december 2002	Projektnummer E70553
	Verksamhetsgren 5. Uppdragsfinansierad verksamhet	
	Delområde 61 Telekrigföring med EM-vapen och skydd	
Författare/redaktör Christer Wigren	Projektledare Gustaf Olsson	
	Godkänd av Mikael Sjöman	
	Uppdragsgivare/kundbeteckning FM	
	Tekniskt och/eller vetenskapligt ansvarig Christer Wigren	
Rapportens titel Realtidssimuleringar av målsökare och motmedel (rök, vattendimma) i en komplex bakgrundsmiljö		
Sammanfattning (högst 200 ord) Realtidssimulering med simuleringsverktyg som används för förberedelser och utvärdering av försök med multispektral skyddsrok (t.ex. multispektral vattendimma) och för kompletteringssimuleringar i syntetiska miljöer beskrivs. Rapporten visar på att en realtidsanpassning av simuleringsverktyg väsentligt kan öka användbarheten av dessa. Hur en realtidsanpassning har/ska gå till diskuterats. Realtidsanpassningen kan delas in i två delar, en del som har med grafikgenerering att göra (innefattar bl.a. motmedelsmodellen) och en del som har med bildbehandling att göra (målföljare/målsökare). Realtidsanpassning av målföljarmodellen innebär bl.a. att simuleringar kan göras med mer avancerade målsökaralgoritmer (t ex multisensormålsökare) och att simuleringarna kan göras direkt mot sensorsignalen.		
Nyckelord Simulering, realtid, målsökare, motmedel, multispektral vattendimma, rök		
Övriga bibliografiska uppgifter	Språk Svenska	
ISSN 1650-1942	Antal sidor: 24 s.	
Distribution enligt missiv	Pris: Enligt prislista	

Issuing organization FOI – Swedish Defence Research Agency Command and Control Systems P.O. Box 1165 SE-581 11 Linköping	Report number, ISRN FOI-R--0694--SE	Report type Methodology report
	Research area code 6. Electronic Warfare	
	Month year december 2002	Project no. E70553
	Customers code 5. Commissioned Research	
	Sub area code 61 Electronic Warfare including Electromagnetic Weapons and Protection	
Author/s (editor/s) Christer Wigren	Project manager Gustaf Olsson	
	Approved by Mikael Sjöman	
	Sponsoring agency FM	
	Scientifically and technically responsible Christer Wigren	
Report title (In translation) Real time simulations of target seekers/trackers and countermeasures (multispectral smoke and waterfog) in a complex background		
Abstract (not more than 200 words) Simulations can be used to prepare and evaluate multispectral smoke (e.g. multispectral waterfog) in field trials and to extend the evaluation using a synthetic environment. An adaptation of the models to real time enhances the usability of the models. How such an adaptation can be made is discussed. The discussion is divided into two parts. One part is about real time graphics (which for instance is used when the countermeasure is modeled) and the other part is about image processing in real time (used in models of target tracker). A real time target tracker can be used to model advance algorithms (i.e. multisensor trackers) and also enables the ability to directly use the signal from the sensor during the field trial.		
Keywords Simulation, real time, target tracker/seeker, countermeasure, multispectral waterfog, smoke		
Further bibliographic information	Language Swedish	
ISSN 1650-1942	Pages 24 p.	
	Price acc. to pricelist	

Innehåll

1	INLEDNING	7
1.1	Bakgrund - simulering av målsökare och motmedel	7
1.2	Realtidssimulering - varför?	7
2	REALTIDSSIMULERING	8
2.1	Grafiksimulering	8
2.2	Optimerade bildbehandlingsbibliotek	12
2.3	Hårdvara för värdering av fältförsök med sensorbaserad målföljningssimulering i realtid	21
3	SAMMANFATTNING OCH SLUTSATSER	22
4	REFERENSER	24

1 Inledning

Denna rapport beskriver hur realtidssimulering kan användas för försök och utvärdering av elektrooptiska motmedel (rök och multispektralvattendimma). För att sätta in det i ett sammanhang ges först en bakgrund hur simulering, i dessa sammanhang, har använts inom projektet VMS/strf.

1.1 Bakgrund - simulering av målsökare och motmedel

Projektet VMS/strf har varit indelat i flera delar varav en del har handlat om motmedel, framförallt i form av multispektral vattendimma. Den multispektrala vattendimman har studerats genom teoretiska studier och fältförsök (fältförsöken har till största del skett inom ramen för studieförsök MUSVAD). Teoretiska studier har inneburit att en datormodell, MUSS, framtagen inom projektet *Tekniska och stridstekniska systemdueller* för att simulera effekten av stridsrök [1], utvecklats för att kunna simulera effekten av multispektral vattendimma. Simuleringsresultat från modellen har använts som ingångsdata till fältförsök vad gäller antal munstycken som behövs och hur dessa skall placeras för att erhålla bästa resultat. Resultat från fältförsök och simuleringar har sedan jämförts och ibland har därigenom önskemål om ny funktionalitet hos datormodellen uppkommit [2].

Fältförsök med vattendimma har förutom att verifiera datormodellen MUSS också använts till att värdera effekten av den multispektrala vattendimman som motmedel mot olika hot framförallt sådana hot som använder bildalstrande sensorer (slutfasstyrd granat, bildalstrande pv-robot). Värderingen har skett genom att registrera bilder och sekvenser både inom IR och inom det visuella våglängdsområdet. Stillastående sensorer på nära håll har används för att få en uppfattning om dimmans täckning av skyddsobjektet som funktion av tiden. Sensorer ombord på helikopter har använts för att registrera närmandeförlopp som ungefärligt motsvarar det som en annalkande hotsensor uppfattar eller ett förlopp som en invisande sensor ser. De registrerade sekvenserna har sedan matats in i målsökarmodeller och simuleringsresultaten har fått ge svar på om den multispektrala vattendimman har någon effekt mot dessa typer av målföljare [3].

En svaghet med fältförsök är att det inte går att på ett kontrollerat sätt variera alla parametrar som kan påverka verkningsgraden av motmedel (i fallet multispektral vattendimma kan dessa parametrar vara vind, kontrast, robothastighet, mm). Det är därför värdefullt att komplettera fältförsök med simuleringar i en syntetisk miljö [4, 5].

1.2 Realtidssimulering - varför?

Vid utvecklingen av de datormodeller som använts inom projektet, för förberedelser av fältförsök och utvärdering av den multispektrala vattendimman, har hög noggrannhet haft högsta prioritet och snabb exekverbarhet har knappt beaktats alls. Detta har gjort att trovärdigheten hos simuleringsresultaten har varit hög men en enskild simulering kan ta allt från flera minuter (en ögonblicksbild från MUSS) till flera timmar (en robotsimulering) vilket gjort att användbarheten i praktiken ibland blivit något begränsad. Genom en realtidсанpassning av datormodeller kan användbarheten hos dessa öka avsevärt (se nedan). Däremot bör de ursprungliga modellerna finnas kvar för att kunna användas till bl.a. valideringssimuleringar.

Datormodeller inom projektet, VMS/strf, har använts till att:

- Förbereda fältförsök (MUSS)
- Utvärdera fältförsök (målföljarmodeller)
- Komplettera resultat från fältförsök med simuleringar i en virtuell miljö.

En realtidsanpassning av vattendimmodellen i MUSS gör det betydligt lättare att testa många olika konfigurationer av vattendimsystemet som förberedelser inför fältförsök. Vid fältförsöken blir det också lättare att testa olika idéer eller att testa vad en given konfiguration kan ge för effekt vid en annan vädersituation. En realtidsanpassning av målföljarmodeller gör det möjligt att koppla en eller flera sensorer till en datormodell av en målsökaralgoritm för att, direkt under fältförsöket, i realtid kunna se effekten av någon åtgärd. Detta tillsammans med ett avancerat och optimerat bildbehandlingsbibliotekⁱ gör det också möjligt att titta på mer avancerade målsökaralgoritmer (t ex multisensoralgoritmer). Robot-simuleringar i realtid kräver förutom realtidsanpassning av motmedels- och målsökarmodellen också en realtidsanpassning av scengenereringsmotorn.ⁱⁱ

Realtidsanpassningen är inte bara användbar till värdering av multispektral vattendimma utan kan också användas för värdering av andra motmedel inom framförallt det elektrooptiska våglängdsområdet (t ex multispektral rök eller facklor).

2 Realtidssimulering

I denna rapport beskrivs ett tänkbart tillvägagångssätt för att åstadkomma realtidsanpassningen för de olika modellerna som nämnts ovan. Beskrivningen kan delas in i två delar. Den ena delen bygger på realtidsgrafik genom Vega/SensorVision och den andra delen bygger på bildbehandlingsbibliotek och bildbehandlingsprocessorer som instickskort i en PC. I detta kapitel beskrivs de två delarna som ingår i realtidsanpassningen. Det mesta av det som diskuteras finns redan färdigt (modeller för facklor och rök, målföljarmodeller i form av korrelation, centroid och retikel) medan andra delar kräver fortsatt arbete (realtidmodell av vattendimma, andra typer av målföljare baserade på exempelvis rosettskannande och kryssdetektor och sensorbaserad målföljning i realtid med hjälp av hårdvara).

2.1 Grafiksimulering

Det har på senare år kommit fram realtidsgrafikmotorer innehållande grafikbibliotekⁱⁱⁱ som kan simulera scenarier både i IR och i det visuella våglängdsområdet [6,7]. Ett sådant grafikbibliotek är Vega från Multigen/Paradigm. Vega kan dels användas ihop med ett användargränssnitt, LynX, som ingår i leveransen av programmet och dels kan Vegas grafikbibliotek användas i egenutvecklade moduler eller helt fristående program. Genom LynX är det lätt att sätta upp ett scenario då det i Vega finns verktyg t ex för att styra en observatör (med hjälp av t ex mus, joystick eller tangentbord), funktioner som håller ett objekt på en konstant höjd över marken, kollisionskontroll, belysningsfunktioner, effekter för att skapa dis, belysning, mm. Till Vega finns också färdiga moduler (bild 1) t ex för att skapa syntetiska IR-bilder baserade på fysikaliska principer. Den IR-modul som finns till Vega kallas SensorVision. Med modulen SensorWorks kan också realistiska sensoreffekter adderas till bilden från SensorVision. SensorWorks kan användas i en visuell och en analytisk mod.

ⁱ Bildbehandlingsbibliotek är ett bibliotek av funktioner för bildbehandling som kan länkas in i egenutvecklade datorprogram.

ⁱⁱ Scengenereringsmotorn kan innehålla grafikbibliotekⁱⁱⁱ och funktioner för att skapa rörliga objekt (se kapitel 2.1).

ⁱⁱⁱ Bibliotek av funktioner som kan användas för att skapa bilder av 3D-objekt i en 3D-bakgrund

Den analytiska moden har jämförbar funktionalitet som den som finns i IGOSS [8]. Beräkningarna i SensorWorks är dock snabbare än i IGOSS men långsammare än om beräkningarna sker i SensorWorks med visuell mod. Alla funktioner som finns i Vegas egna användargränssnitt, LynX, finns också som anropbara C-funktioner för implementering i egna program och moduler. För egenutvecklade moduler går det att skapa egna användargränssnitt i LynX (bild 2).

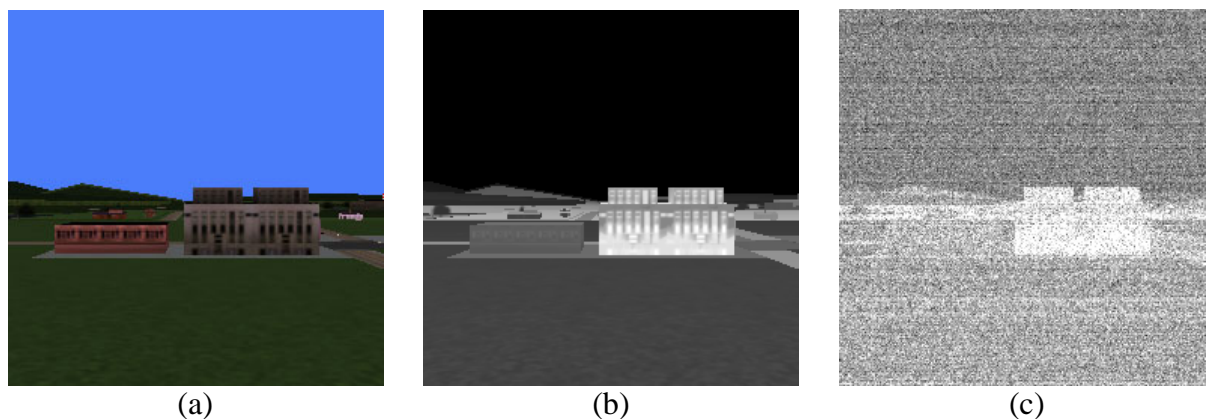


Bild 1 Scenario genererat i Vega/SensorVision/SensorWorks. (a) Visuell - enbart Vega, (b) IR - Vega med SensorVision, (c) IR med sensor prestanda - Vega med SensorVision och SensorWorks.

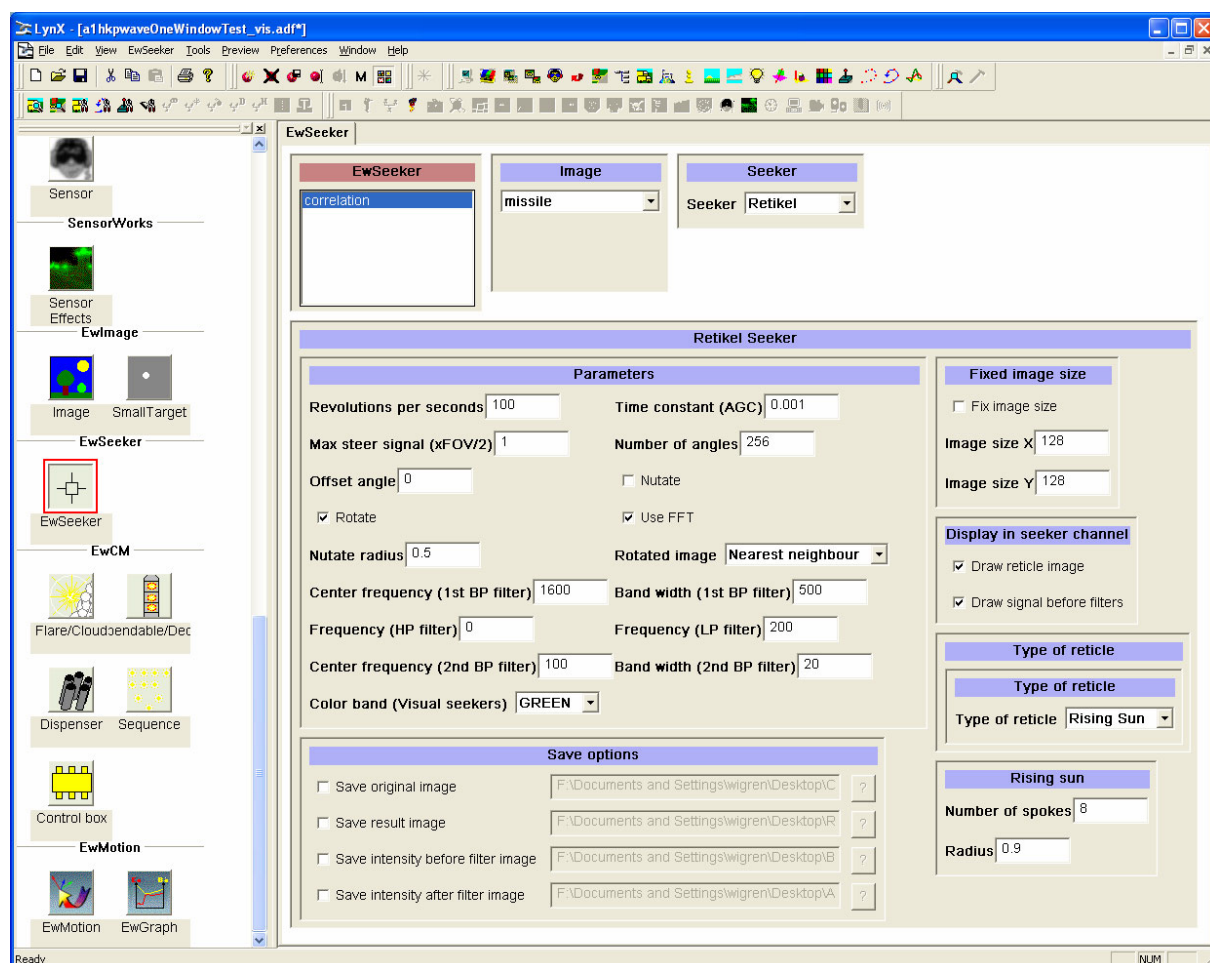


Bild 2 LynX med användargränssnitt till egenutvecklad EwSeeker modul.

Till Vega har det inom FOI utvecklats flera egna moduler som behövs för att kunna göra telekriksduellsimuleringar [9]. Dessa är moduler för motmedel (multispektral vattendimma, multispektral rök samt facklor), moduler för att extrahera bilder som sedan kan sparas som en bildsekvens eller sändas vidare till en målföljarmodul. Det finns flera målsökartyper framtagna. Många av dessa målföljarmodeller togs fram genom att, utan större modifiering, kapsla in kod från tidigare målföljarprogram i kod som kan kommunicera med Vega. Exempel på sådana målföljartyper är korrelation och centroid. Andra har krävt större modifieringar (retikel) och krävt att optimerade bildbehandlingsbibliotek (se fotnot ⁱ, sidan 8) använts (en beskrivning av retikelmodellen finns i nästa avsnitt) för att erhålla realtids- eller nära realtidsprestanda. En stor del av denna modulutveckling har utförts inom projektet *Stridstekniska systemdueller*.

Bild 3 visar egenutvecklade moduler som kopplats till Vega/SensorVision för telekriksduellsimuleringar. Dessa moduler kan delas in i två grupper. Den ena gruppen behövs för att kunna styra roboten och den andra har med målets signatur att göra och används för att simulera effekten av motmedel i bilden. Robotens styrning bör baseras på vad sensorn ser, därför behövs en omvärldsbeskrivning som extraheras från Vega/SensorVision i form av en bild (*Image*). När bilden av målet blir mindre än ett bildelement kan intensiteten från målet bli felaktig eftersom 3D-grafikmotorn i Vega bara tittar på vad som finns mitt i varje bildelement. Om ett objekt helt ryms inom ett bildelement syns inte objektet om det inte täcker centrum av bildelementet, alternativt blir intensiteten i bildelementet lika med intensiteten från någon liten del på objektet. För att erhålla rätt intensitet kan den del av bilden där målet befinner sig därför behöva förstöras och medelvärdesbildas och sedan matas tillbaka i bilden (*SmallTarget*) innan den skickas vidare till målsökaren. Flera olika målföljare (korrelation, centroid, och retikel - bygger till stor del på tidigare fristående målföljarmodeller [10], [11], [12]) - finns implementerade i målsökarmodulen (*EwSeeker*): *EwSeeker* utnyttjar någon form av bildbehandling för att extrahera målets position i bilden. Efter att målets position bestämts skickas informationen vidare till robotdynamikmodulen (*EwMotion*) som bestämmer hur roboten skall röra sig för att träffa målet.

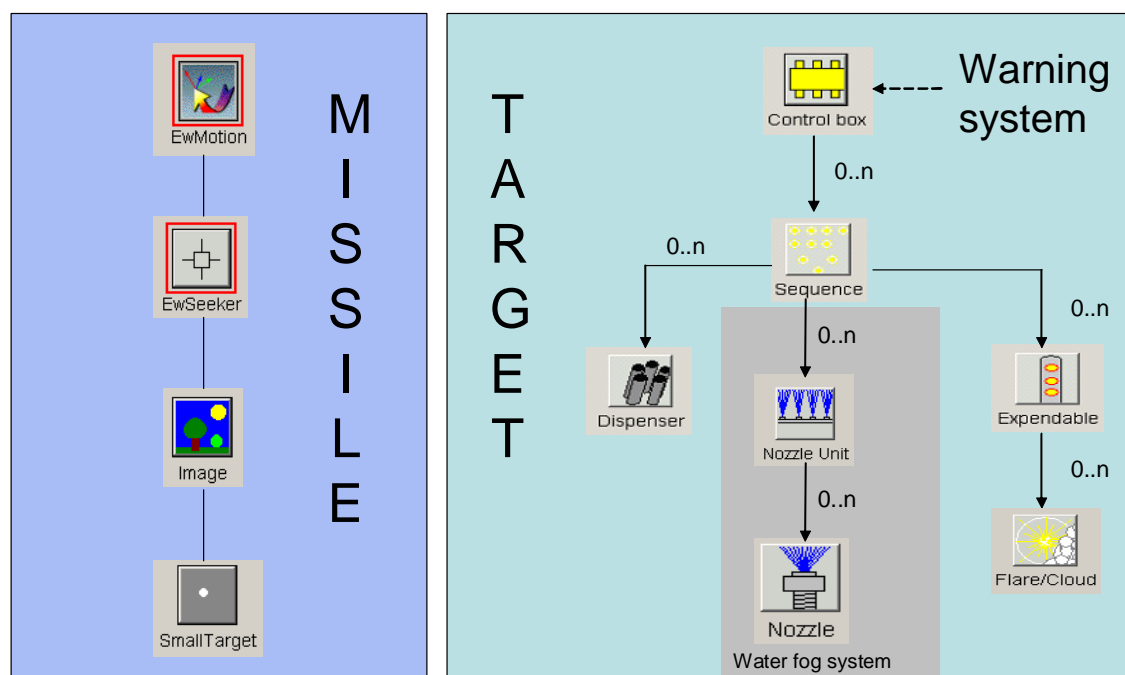


Bild 3 Egenutvecklade moduler i Vega för telekrigduellsimuleringar.

Den bild som roboten utnyttjar för att styra mot målet genereras av Vega eventuellt med SensorVision eller SensorWorks. Det användaren måste göra är att se till att Vega har en bra geometrisk och textuell beskrivning av hur målet ser ut. För att beskriva målets utseende i det infraröda området, och möjliggöra att signaturen ändras efter bl.a. tid på dygnet och våglängdsområde, måste målets yta materialklassas så att SensorVision kan räkna ut målets signatur dynamiskt. Materialklassning av ett objekt kan för en erfaren person typiskt ta ett par dagar under förutsättning att allt underlag (t.ex. i form av mätunderlag och 3D-modeller) finns tillgängligt.

För att generera effekten av motmedel i form av facklor, rök och multispektral vattendimma finns inga färdiga moduler i Vega/SensorVision. Den andra gruppen av egenutvecklade moduler (*EwCM*) hanterar denna generering. En submodul i *EwCM* har till syfte att starta motmedelsgenerering enligt något givet kriterium (*Control box*). De kriterier som kan användas är (1) en interaktion från användaren genom en tangenttryckning, (2) en tidsperiod från det att simuleringen startar och (3) ett minsta avstånd mellan robot och mål. En startsignal skickas från *Control box* till en annan submodul (*Sequence*) i *EwCM*. I *Sequence* bestäms ett godtyckligt antal tidpunkter från det att startsignal givits då kastaren (*Dispenser*) skall skicka iväg sina laddningar (facklor eller rökgranater). Rök eller fackelgranaternas utskjutningshastighet relativt objektet och hur de beter sig i lufthavet innan de exploderar definieras i *Expandable*. Efter att granaterna exploderat bildas rök eller facklor vars intensitet och rörelse beskrivs av *Flare/Cloud*. Flera objekt kan skapas av samtliga moduler. En *Control box* kan kontrollera en eller flera *Sequence*. *Sequence* kan styra en eller flera *Dispenser*. *Dispenser* bestämmer positionen på kastaren relativt skyddsobjektet och kan innehålla flera olika typer av granater. Slutligen kan en granat (*Expandable*) innehålla flera subdelar (facklor eller rök, *Flare/Cloud*) av samma eller olika typ.

Den fristående motmedelsmodellen, MUSS, beskriver rök och facklor med hjälp volymelement och effekten av motmedel för en sensor simuleras genom att integrera strålvägar genom röken, dimman eller facklan. För exempelvis stora rök- eller dimmoln på

nära håll leder detta till ganska krävande beräkningar som tar lång tid. I realtidssammanhang är därför inte detta en framkomlig väg med dagens persondatorer utan någon form av approximativ lösning måste användas. I realtidssfallet beskrivs därför motmedlet med hjälp av plana ytor som innehåller en bild med information om motmedlets signatur och transparens som funktion av position i planet. Dessa plan har, för en betraktare utanför motmedlet, samma transmission och signatur som volymentelementet i MUSS. Nackdelen är att när objektet befinner sig inom t ex ett rökmoln kommer röken att bli fullständigt transparent för delar av objektet som sticker ut utanför de semitransparenta ytorna, delar som befinner sig bakom kommer att betraktas som om det befann sig bakom hela rökmolnet. I verkligheten och i MUSS försämras transmissionen gradvis till delar som befinner sig längre in i rökmolnet. Delvis kan man komma ifrån detta genom att bygga upp lager av semitransparenta plan kring objektet. Transmissionen kommer i detta fall att ändras stegvis när avståndet till centrum av rökmolnet ändras och genom att öka antalet transparenta skikt kan noggrannheten öka.

När det gäller multispektral vattendimma måste en lösning med flera transparenta skikt väljas för realtidstillämpningar eftersom skyddsobjektet ofta befinner sig inne i dimmolnet. I detta fall kastas inte heller granater iväg med en kastare utan genereras i munstycken som sitter fast monterade på eller i anslutning till objektet. Här är tanken att modulen *Nozzle* beskriver en typ av munstycke vad gäller vattenmängd, droppstorlekar, spridningsvinklar, mm. Modulen *Nozzle Unit* beskriver hur dessa munstycken kopplas samman: antal, inbördesavstånd, osv. *Nozzle Unit* styrs fortfarande av modulen *Sequence* även om bara en tidpunkt är aktuell att ge i detta fall (dvs. hur snabbt efter varning eller manuell knapptryckning är systemet trycksatt och kan börja generera multispektral vattendimma).

2.2 Optimerade bildbehandlingsbibliotek

Modeller för enklare typer av bildalstrande målföljare (exempelvis av typen centroid eller korrelation med en liten korrelationskärna) kan utan större svårigheter fås att fungera i en realtids- eller nära realtidsmiljö utan att använda optimerade bildbehandlingsbibliotek. Målsökarmodeller med avancerad bildbehandling eller där avancerad bildbehandling krävs för att efterlikna målföljarens funktion ställer däremot stora krav på att optimerad programkod används. Retikelmålsökare har en enkel funktion men avancerad bildbehandling krävs för att efterlikna målsökarens funktion om hänsyn skall kunna tas till bakgrundsklotter. Denna typ av målsökare utnyttjar en roterande eller nuterande skiva (retikel) placerad i fokalplanet av målsökarens optik. Retikeln har fält med olika transparens (bild 4) vilket gör att signalen som detektorn ser bakom retikeln moduleras beroende på var målet befinner sig. Detta gör att signalen som funktion av retikelns vinkel kan, efter signalbehandling, ge information om i vilken riktning, i förhållande till målsökaren, målet befinner sig (bild 5). För att simulera denna funktion kan en retikelbild överlagras ingångsbilden och alla bildelement integreras som funktion av retikelns vinkel (bild 6). Olika signalsvar ges beroende på var målet befinner sig (bild 7).



Bild 4 Exempel på retiklar i en retikelmålsökare (vit motsvarar fullständig transparens, svart - ogenomskinlig och grå - semitransparent).

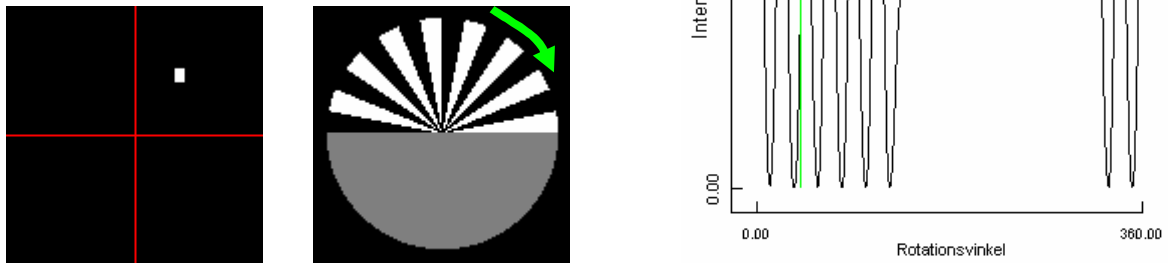


Bild 5 Signal från roterande retikel som anger var målet befinner sig. Ingångsbild till vänster, retikelbild i mitten och signal som funktion av rotationsvinkel till höger.

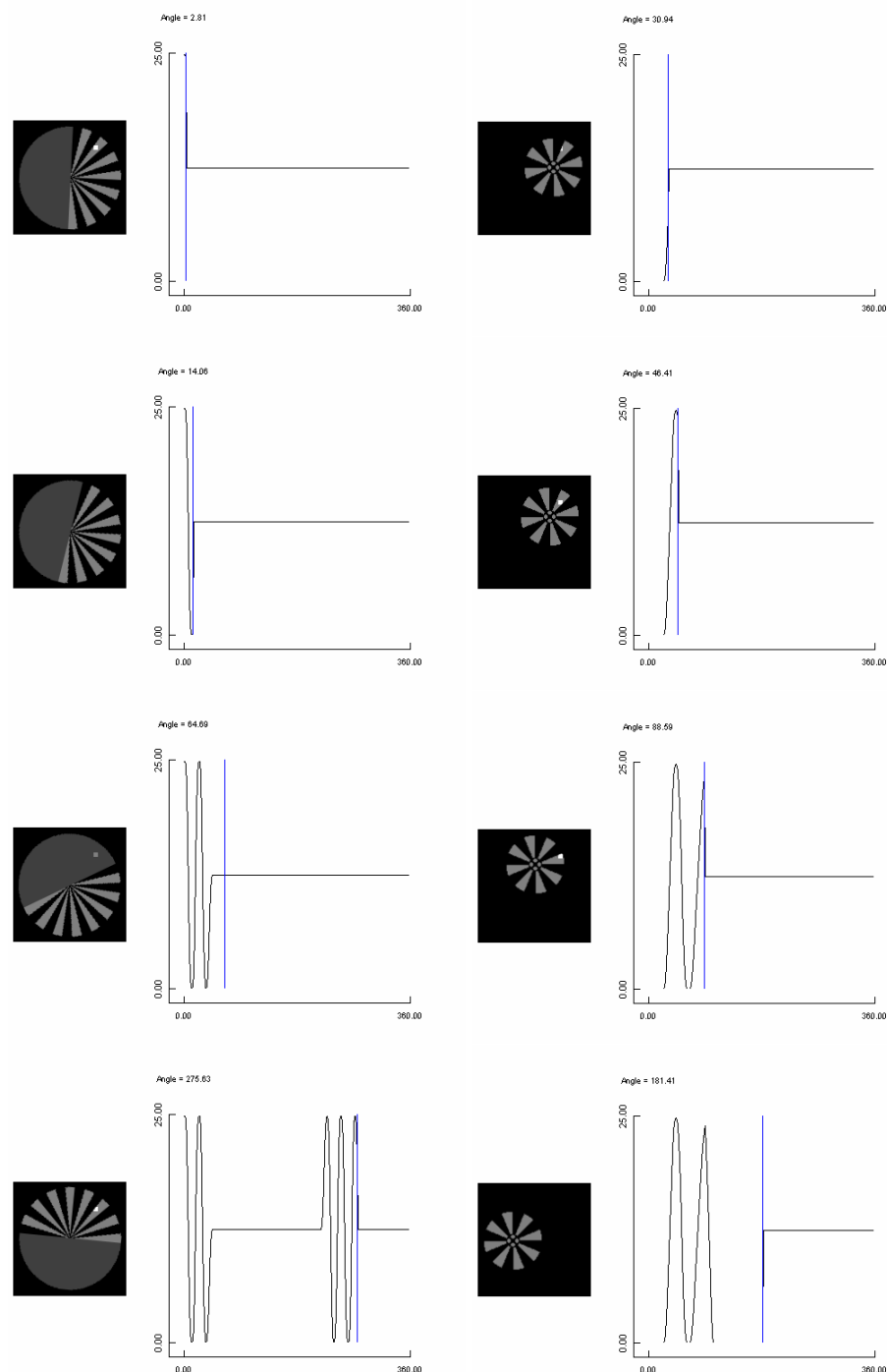
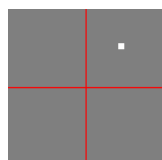


Bild 6 Illustration som visar hur retikelmålsökarens signal byggs upp. Bakgrundsbilden finns längst upp i bilden. I kolumnen till vänster visas en roterande retikel överlagrad bakgrundsbilden som funktion av tid. Kolumnen till höger om denna visar hur signalen från denna retikelmålsökare byggs upp. Nästa kolumn visar en nuterande retikel överlagrad bakgrundsbilden som funktion av tid. Den sista kolumnen visar den nuterande retikelns signal.

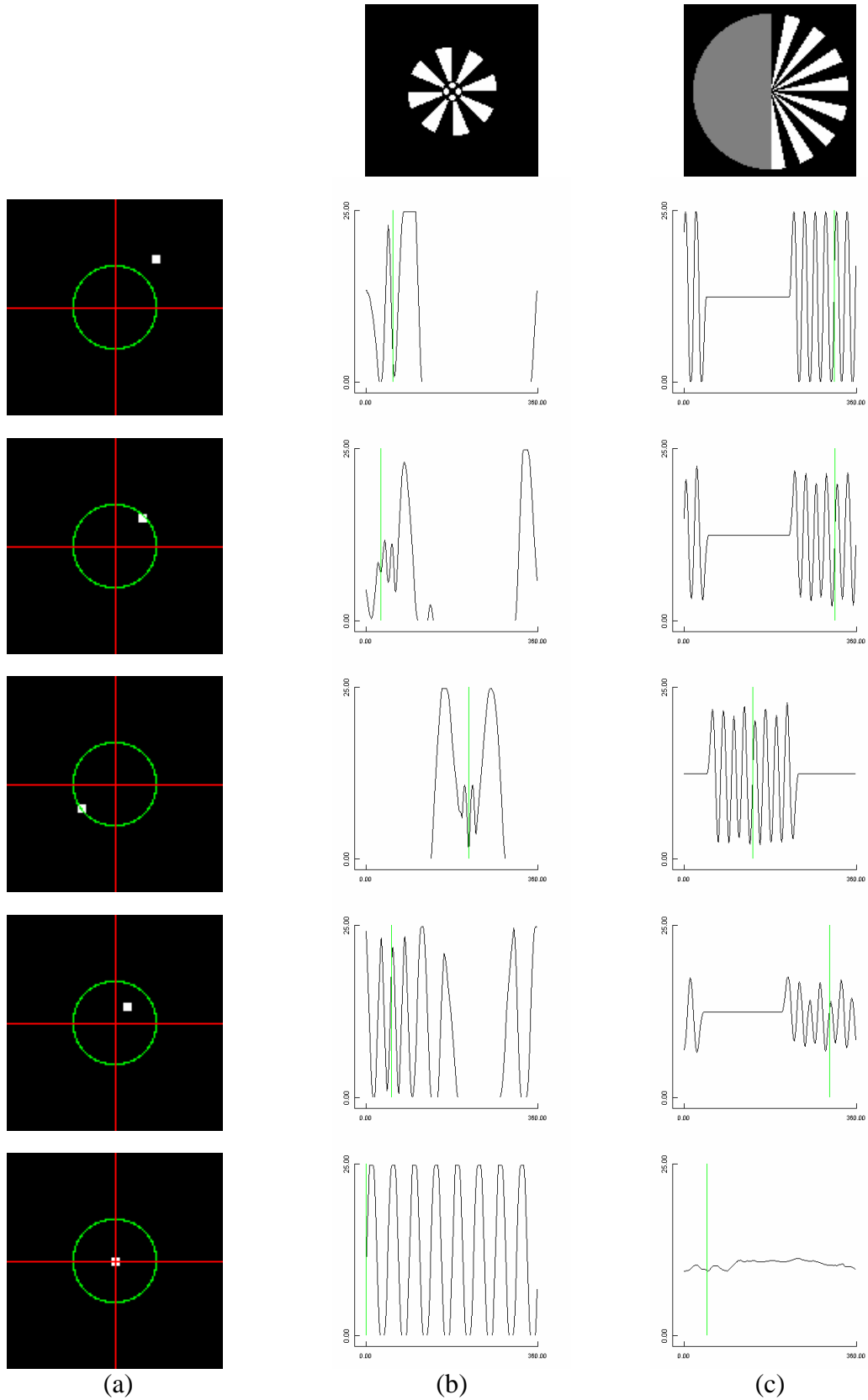


Bild 7 Visar (a) målets position i bilden (den gröna cirkeln visar nutationsradien för den nuterande retikeln). Retikelsignalen från dessa bilder visas i (b) för en nuterande retikel och i (c) för en roterande retikel. Längst upp visas bilderna av retiklarna.

I en realtidsmodell skall retikeln rotera med ca 100 Hz och för att erhålla någorlunda bra signalsvar är 5° vinkelsteg ett absolut minimum. Bilden av scenen och retikeln bör vara minst 128x128 bildelement stor. Dessutom kanske retikelbilden måste roteras eller translateras och det antal matematiska operationer som krävs för denna process är alldeles för stort för att dagens PC skall kunna hantera detta i realtid. Ett matematiskt alternativ till denna integrationsmetod som funktion av vinkel är att använda Fouriertransformer. Genom att utnyttja Fouriertransformer kan två bilder faltas med varandra. Om minst en av bilderna i faltningsprocessen är liten sker faltningsprocessen lämpligast genom att som ovan multiplicera bilderna med varandra, flytta den ena bilden till nästa position och göra om processen. Med större bilder går det däremot mycket snabbare att först Fouriertransformera båda bilderna sedan multipliceras varje bildelement i de Fouriertransformerade bilderna med varandra och lagras i en resultat bild. En invers Fouriertransform görs sedan på resultatbilden och värdet i varje bildelement motsvarar nu resultatet av en bildmultiplikation efter att den ena bilden har translaterats så att ena hörnet på denna bild ligger i denna position (med förbehandling innan Fouriertransformen erhålles resultatet av att centrum på ena bilden har hamnat i denna position). För att erhålla signalsvaret från en nuterande retikel läses nu värdet av för lämpliga bildelement längs en radie från centrum av resultatbilden som motsvarar retikelns nuterande rörelse (bild 8 och 9).

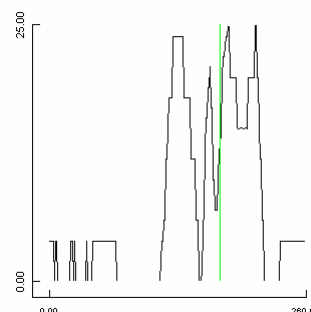
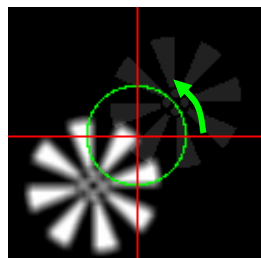
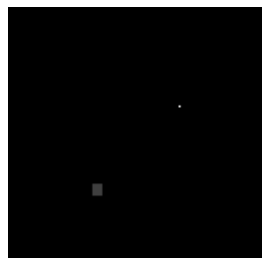


Bild 8 Bilden av den nuterande retikeln visas längst upp. Målbild med ett stort (5x5 bildelement) och ett litet (1 bildelement) mål visas längst ner till vänster. Dessa två bilder har faltats med varandra (mitten). Signalsvaret (höger) kan sedan läsas av längst den gröna linjen (cirkeln).

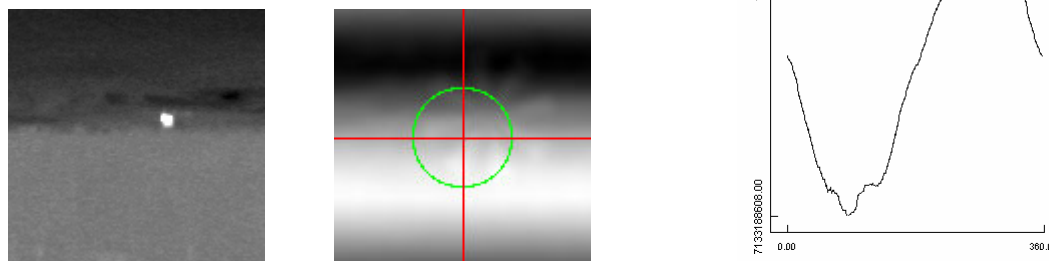
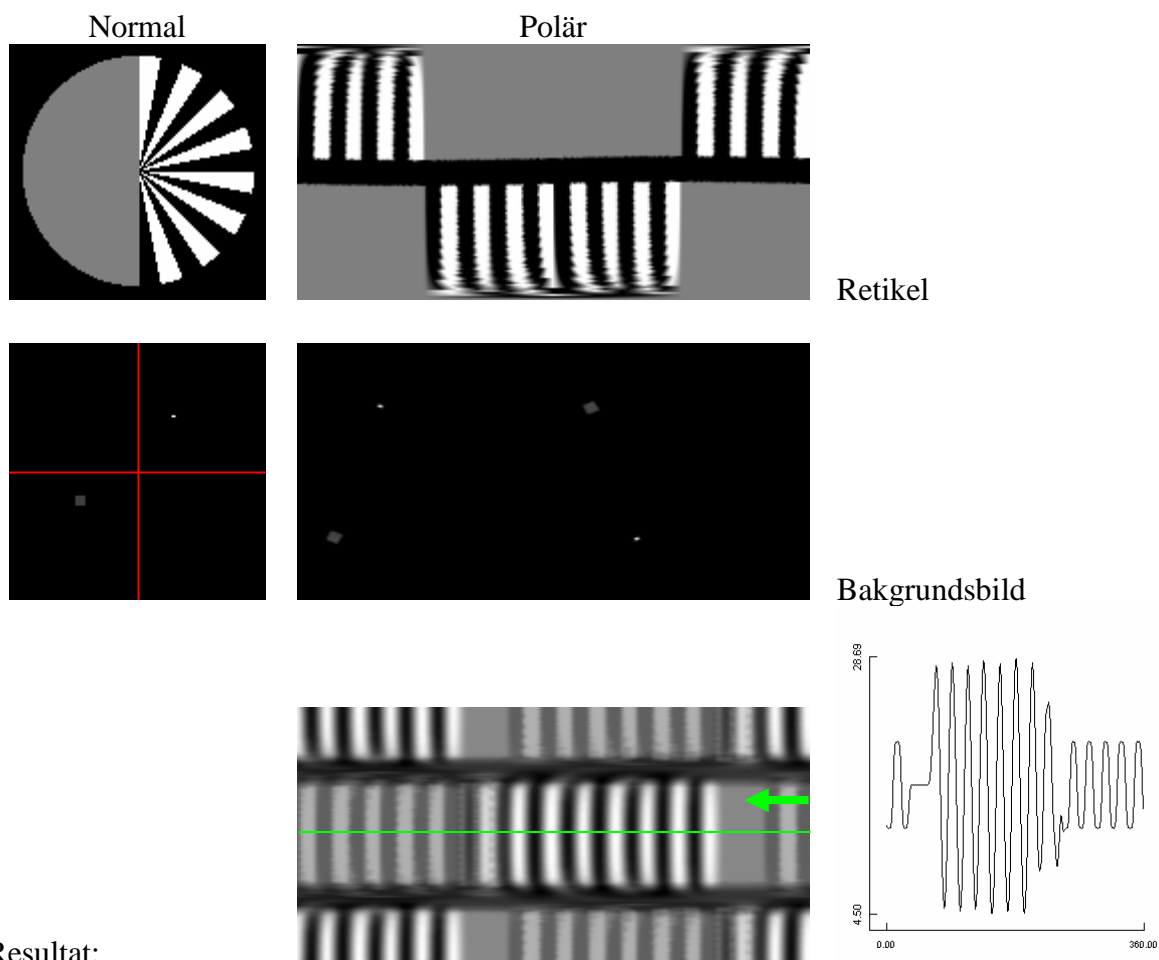


Bild 9 Illustration över hur faltningsprocessen kan utnyttjas för att erhålla signalsvaret från en nuterande retikel. Samma retikel som i bild 8 men nu med en verklig bakgrundsbild. På samma sätt som tidigare läses signalen av längs den gröna linjen (cirkeln).

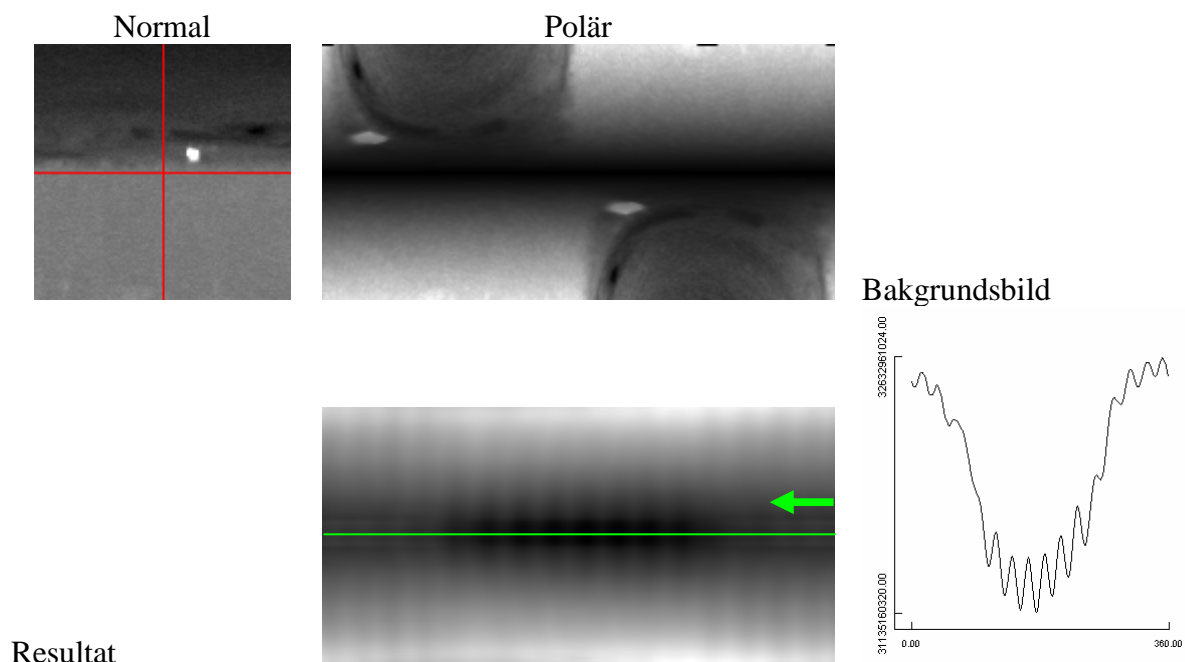
Med en roterande retikel blir det hela lite mer komplicerat än vid nutation. För att kunna räkna ut signalsvaret vid rotation med hjälp av en faltningsmetod måste först bilderna transformeras till polära koordinater. Faltning med hjälp av Fouriertransformer antar att alla bilder är cykliska^{iv}. Detta är inget problem i fallet med nuterande retikel så länge som hela retikeln befinner sig inom synfältet. I fallet med polära koordinater är det svårare men om vi genererar polära bilder med både negativa och positiva värden på radien och låter vinkeln gå från 0 till 360° fungerar det (bildelementet i position (r, θ) är ekvivalent med bildelementet i position $(-r, \theta + 180^\circ)$). För att kunna använda FFT (snabbt sätt att beräkna Fouriertransformen för diskreta funktioner) måste bilderna vara av storleken 2^n i båda ledderna (där n är ett heltal). För att intensiteter skall bli riktiga måste den polära bakgrundsbilden normaliseras eftersom antalet bildelement för små radier är mycket större i den polära bilden jämfört med antalet bildelement i den rektangulära bilden.

^{iv} Bilden är en enhet i ett repeterbart mönster. Det vill säga, vid höger kant av bilden börjar bilden om med vänster kant, vid nedre kanten av bilden börjar bilden om med övre kant, osv.



Resultat:

Bild 10 Illustration över hur faltningsprocessen kan utnyttjas för att erhålla signalsvaret från en roterande retikel. Bilden av retikeln i rektangulära koordinater (längst upp till vänster) och polära koordinater (längst upp i mitten). Bakgrunden visas i mitten raden både i rektangulära och polära koordinater. När de polära bilderna av bakgrunden och retikeln faltas med varandra erhålles resultatet i mittenkolumnen längst ner. Signalsvaret från retikel målsökaren kan sedan läsas av längs den gröna linjen (signalen visas längs ner till höger).



Resultat

Bild 11 Illustration med verklig bild över hur faltningsprocessen kan utnyttjas för att erhålla signalsvaret från en roterande retikel. Samma retikeltyp som i bild 10 används men nu med en verklig bakgrund..

Ett exempel på ett optimerat programbibliotek som klarar av att beräkna rotationer, translationer, FFTer, transformationer mellan rektangulära och polära koordinater är MIL (matrox imaging library) från matrox [13]. Detta programbibliotek har den fördelen att det förutom att kunna användas fristående också är kompatibelt med en serie instickskort som i vissa fall kan snabba upp beräkningarna (tabell 1) och också möjliggör inläsning av bilder från externa sensorer (t ex en eller flera IR-kameror).

Tabell 1 Jämförelse mellan MIL med och utan hårdvaruacceleration. (Hämtad från referens 13).

MIL/ActiveMIL Benchmarks

The following benchmarks provide a performance overview for a range of imaging operations running on different platforms. A brief description of all functions, parameters and images used are included. Note that the benchmarks assume full CPU and memory bandwidth (i.e., no other system activity) and include system overheads.

Note: Operations executed on 512 x 512 images.

	Pentium 4 2.5 GHz 533 MHz FSB PC800 SDRAM	Athlon™ XP 2000+ 240 MHz FSB PC2100 DDR SDRAM	Pentium® III 1 GHz 100 MHz FSB PC100 SDRAM	Celeron® 800 MHz 100 MHz FSB PC100 SDRAM	GematekPlus 1.25 GHz FSB PC133 SDRAM
Image Processing					
Point-to-point Add two 8-bit images and store results in an 8-bit destination image.	0.51 ms	0.96 ms	1.7 ms	2.2 ms	1.1 ms
Edge Detection (sobel) Perform an edge detection (sobel) on an 8-bit source image and store results in an 8-bit destination image.	1.3 ms	1.5 ms	3.1 ms	3.6 ms	2.0 ms
Convolution (3 x 3) Perform a general 3 x 3 convolution with arbitrary coefficients on an 8-bit source image and store results in an 8-bit destination image. Results are saturated.	1.3 ms	1.8 ms	2.8 ms	3.2 ms	1.7 ms
Convolution (5 x 5) Same as above except with a 5 x 5 kernel.	3.5 ms	4.3 ms	7.7 ms	8.8 ms	3.3 ms
Convolution (11 x 11) Same as above except with a 11 x 11 kernel.	16.2 ms	16.8 ms	31.3 ms	36.6 ms	10.9 ms
Erosion/Dilation (3 x 3, predefined, binary) Perform a binary erosion/dilation on a 1-bit source image using a predefined 3 x 3 structuring element and store results in a 1-bit destination image.	0.046 ms	0.077 ms	0.14 ms	0.35 ms	0.36 ms
Erosion/Dilation (3 x 3, predefined, grayscale) Same as above except perform a grayscale operation.	0.54 ms	1.5 ms	2.3 ms	2.6 ms	1.7 ms
Erosion/Dilation (3 x 3, user-defined, binary) Perform a binary erosion/dilation on a 1-bit source image using an arbitrary 3 x 3 structuring element and store results in a 1-bit destination image.	0.17 ms	0.26 ms	0.47 ms	0.73 ms	0.65 ms
Erosion/Dilation (3 x 3, user-defined, grayscale) Same as above except perform a grayscale erosion/dilation operation.	1.1 ms	2.4 ms	2.9 ms	3.1 ms	1.7 ms
Erosion/Dilation (5 x 5, user-defined, binary) Perform a binary erosion/dilation on a 1-bit source image using an arbitrary 5 x 5 structuring element and store results in a 1-bit destination image.	0.71 ms	1.0 ms	2.0 ms	2.6 ms	0.66 ms
Erosion/Dilation (5 x 5, user-defined, grayscale) Same as above except perform a grayscale erosion/dilation.	2.2 ms	5.8 ms	4.7 ms	5.4 ms	4.2 ms
Histogram Calculate the histogram of an 8-bit source image and store result in a 32-bit buffer.	0.51 ms	0.68 ms	2.3 ms	2.8 ms	1.8 ms
LUT map Perform a point-to-point LUT mapping operation for an 8-bit source image and store results in an 8-bit destination image.	0.40 ms	1.1 ms	2.6 ms	2.6 ms	1.3 ms
Lossless JPEG Compression (monochrome) Perform lossless JPEG compression on an 8-bit source image and store results in an 8-bit destination image.	4.0 ms	5.0 ms	8.6 ms	9.3 ms	2.7 ms
Lossy JPEG Compression (monochrome) Same as above except perform lossy JPEG compression.	2.6 ms	3.2 ms	4.9 ms	5.9 ms	7.6 ms
Lossy JPEG Compression (color) Same as above except on a 24-bit (packed RGB) source image and store results in a 16-bit (YUV16) destination image.	4.9 ms	9.0 ms	12.2 ms	14.6 ms	n/a
Lossless JPEG2000 Compression (monochrome) Perform lossless JPEG2000 compression on an 8-bit source image and store results in an 8-bit destination image.	109.7 ms	128.3 ms	122.9 ms	280.5 ms	n/a
Lossy JPEG2000 Compression (monochrome) Same as above except perform lossy JPEG2000 compression.	78.0 ms	91.4 ms	94.8 ms	195.8 ms	n/a
Rotate (55°) Rotate by 55° an 8-bit source image and store results in an 8-bit destination image.	2.3 ms	3.4 ms	5.8 ms	6.7 ms	4.8 ms
Warp Polynomial Warping using a first-order polynomial mapping with nearest neighbor interpolation on an 8-bit source image and store results in an 8-bit destination image.	5.3 ms	6.2 ms	8.0 ms	9.3 ms	3.6 ms
Bayer Filter Apply Bayer filter on an appropriate 8-bit source image and store results in a 32-bit (packed RGB) destination image.	1.8 ms	2.4 ms	3.9 ms	5.4 ms	n/a

Resultat från simuleringar med retikelmodellen finns redovisad i tabell 2. Vid dessa simuleringar användes en färdiginspelad sekvens (med 272x136 bildelement). Bilderna i sekvensen storleksförändrades vid beräkningarna för att erhålla den efterfrågade storleken. En retikelmålsökare har typiskt en rotationsfrekvens på ca 100 Hz. Detta betyder att det är endast för en bildstorlek på 64x64 (64 och 128 vinklar per varv) och en nuterande retikel som målsökarmodellen går i realtid på den dator som användes vid testet. Testet visar dock att, för många tillämpningar där en användarens interaktion i realtid inte är kritisk, är modellen tillräckligt snabb (även för större bilder och fler vinkelsteg, både för nuterande och roterande retikel) och det bör t.ex. inte vara något problem att använda den vid verkansdiagramberäkningar där tusentals simuleringar måste göras.

Tabell 2 Resultat från simuleringar med retikelmodellen mot färdiginspelad sekvens. Tabellen visar uppnådd bildsekvens för olika typer av operationer. Vid uppspelning görs ingen målsökarberäkning på sekvensen, Nut64 motsvarar en målsökare med roterande retikel och 64 vinkelsteg i beräkningen, Rot256 motsvarar en roterande retikel med 256 vinkelsteg. Bilderna i sekvensen har en fix storlek (273x136) och omvandlas i programmet till given storlek (64x64, 128x128, 256x256 eller 512x512). Simuleringarna är gjorda på en dator med dubbla XEON processorer med 2,2 GHz klockfrekvens och 1,0 GB RAM.

Uppspelning

Bildfrekvens [Hz] vid 64x64
Bildfrekvens [Hz] vid 128x128
Bildfrekvens [Hz] vid 256x256
Bildfrekvens [Hz] vid 512x512

194,1
161,9
102,5
41,1

Nutation

Vinkelsteg

Metod

Bildfrekvens [Hz] vid 64x64
Bildfrekvens [Hz] vid 128x128
Bildfrekvens [Hz] vid 256x256
Bildfrekvens [Hz] vid 512x512

Metod	64/varv		128/varv		256/varv		512/varv	
	Med FFT	Utan FFT	Med FFT	Utan FFT	Med FFT	Utan FFT	Med FFT	Utan FFT
Bildfrekvens [Hz] vid 64x64	133,7	29,3	116,9	16,4	97,2	8,6	72,5	4,3
Bildfrekvens [Hz] vid 128x128	79,7	15,7	78,9	8,3	71,0	4,3	55,7	2,2
Bildfrekvens [Hz] vid 256x256	34,0	4,8	32,2		30,9		28,0	
Bildfrekvens [Hz] vid 512x512	8,5				8,3		8,1	

Rotation

Vinkelsteg

Metod

Bildfrekvens [Hz] vid 64x64
Bildfrekvens [Hz] vid 128x128
Bildfrekvens [Hz] vid 256x256
Bildfrekvens [Hz] vid 512x512

Metod	64/varv		128/varv		256/varv		512/varv	
	Med FFT	Utan FFT	Med FFT	Utan FFT	Med FFT	Utan FFT	Med FFT	Utan FFT
Bildfrekvens [Hz] vid 64x64	54,0	55,4	49,2	32,2	40,5	17,5	30,7	7,6
Bildfrekvens [Hz] vid 128x128	17,9	21,7	17,0	11,6	15,2	5,9	12,6	3,0
Bildfrekvens [Hz] vid 256x256	5,0	4,7	4,8	2,4	4,6		4,1	
Bildfrekvens [Hz] vid 512x512					1,2		1,1	

2.3 Hårdvara för värdering av fältförsök med sensorbaserad målföljningssimulering i realtid

Under fältförsök används ofta en på förhand uppgjord försöksplan bestående av ett antal olika prov som skall besvara de frågor som fanns innan försöken. Det är inte ovanligt att det efter analys av provresultaten uppkommer ytterligare frågor som kräver ytterligare fältförsök. Utvärdering av försöken direkt under fältförsöken kan öka effektiviteten så att alla frågor, genom mindre modifieringar av försöksplanen, kan besvaras under ett och samma fältförsök. Fältförsök utföres dock ofta under viss tidspress och det är därför inte alltid möjligt att hinna med utvärdering i anslutning till enskilda prov under försöken. Som nämnts i inledningen används ofta målföljarsimuleringar på registrerade sekvenser för att utvärdera motmedlets effektivitet och om registreringsutrustning kan kopplas direkt till en målföljarmodell kan omedelbara utvärderingsresultat erhållas. Detta kan också vara en metod för att lära sig mer

om målföljarprinciper. Exempelvis i en multisensor konfiguration där två sensorer (t ex en IR-sensor i 3-5 μm och en i 8-12 μm området) är kopplade till en målsökarmodell kan parametrar för att vikta de två sensorernas information justeras i realtid för att erhålla bästa mållåsning med och utan motmedel. Givetvis skulle denna metod även kunna användas för att under försök optimera motmedlets effekt mot denna typ av målföljare.

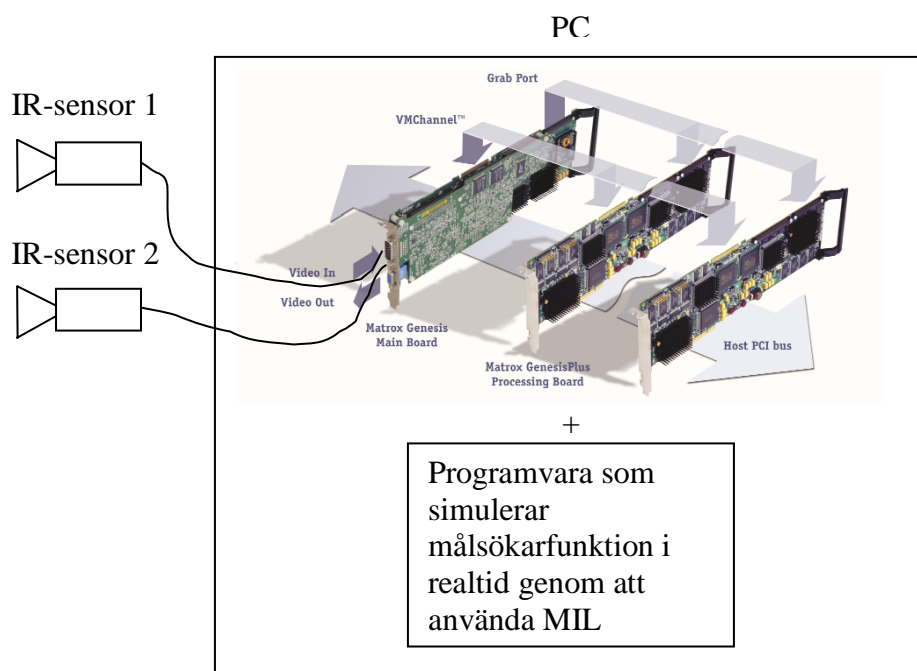


Bild 12 Tänkbar multisensorkonfiguration av ett målföljarsystem. Bilden visar att fler än ett Matrox GenesisPlus Processing Board [13] (i bilden finns två sådana kort) kan användas för att förbättra prestanda. Hårdvaran kombineras med programvara som simulerar målsökarfunktion i realtid genom att använda matrox imaging library (MIL).

3 Sammanfattning och slutsatser

För att förbereda och utvärdera motmedel är simuleringsverktyg värdefulla. Rapporten har beskrivit de simuleringsverktyg som använts inom projektet VMS/strf för förberedelser inför fältförsök med multispektral vattendimma, för utvärdering av försöksresultaten och för kompletteringssimuleringar i syntetiska miljöer. Rapporten har visat på att en realtidsanpassning av simuleringsverktygen väsentligt kan öka användbarheten av dessa. Hur denna realtidsanpassning har/ska gå till har också diskuterats. Realtidsanpassningen kan delas in i två delar, en del som har med grafikgenerering att göra (innefattar bl.a. motmedelsmodellen) och en del som har med bildbehandling att göra (målföljare/målsökare). Realtidsanpassning av målföljarmodellen innebär bl.a. att simuleringar kan göras med mer avancerade målsökaralgoritmer (t ex multisensormålsökare) och att simuleringarna kan göras direkt mot sensorsignalen.

En del av de idéer som presenterats i denna rapport har ännu ej realiserats fullt ut. Detta gäller t.ex. simulering av vattendimma i realtid och den hårdvarubaserade simuleringen i realtid som beskrivs i kapitel 2.3. Realtidsmodeller för båda dessa skulle vara mycket betydelsefulla vid fortsatta försök med multispektral vattendimma. Hårdvarubaserad målföljarsimulering i realtid skulle dessutom vara värdefullt vid andra typer av försök med

motmedel eller vid tester av principer för avancerade målföljare, t.ex. baserade på multispektrala effekter.

Modeller för simulering i syntetisk miljö kommer till sin fulla rätt först när de kopplats samman till en applikation. Exempel på en sådan applikation där modeller från *EwSim* används är en modell för att studera fackelfällningstaktik från helikopter som tagits fram inom studien *VMS för helikopter* (HKP99352S). Andra typer av applikationer skulle kunna vara att koppla samman modeller i olika våglängdsområden (visuellt, IR, radar, etc.). För att det senare ska vara realiserbart krävs att modeller för våglängdsområden utanför det visuella och det infraröda tas fram. Framtagning av motmedelsmodeller för laserstörning har påbörjats men det krävs fortsatt arbete innan det kan användas i duellsimulerings-sammanhang. Applikationer med en människa i loopen skulle också kunna vara av intresse att studera. I en sådan applikation kan inverkan av mänskliga beslut tas med i simuleringsresultatet. Alla dessa exempel kräver eller har nytta av realtidsprestanda och är sådant som bör vara av intresse för dagens och framtidens försvar och det är därför viktigt att arbetet med realtidsanpassade modeller fortsätter.

4 Referenser

- [1] Lars Tydén, *MUSS, Multi Spectral Smoke Model*, FOA-rapport FOA-R--97-00576-616--SE, November 1997.
- [2] Lars Tydén, *Lägesrapport modelleringsprogrammet MUSS*, FOI MEMO, Dnr: 01-3916
- [3] FOA och FOI rapporter: FOA-RH--99-00440-616, FOA-RH--00-00534-616, FOI-RH--0090--SE, och Årsrapporter från studieförsök MUSVAD 1997-2000 utgivna av SkyddC.
- [4] Ulrika Andersson, Annelie Tonnvik, *Simulering av störning mot slutfasstyrd granat.*, FOI-rapport FOI-R--0133--SE, April 2001
- [5] Christer Wigren, *Pansarvärnsrobot mot motmedelsskyddade stridsfordon - en simuleringsstudie.*, FOI-rapport, FOI-R--0261--SE, November 2001
- [6] <http://www.cg2.com>
- [7] <http://www.multigen.com>
- [8] Christer Wigren, *IGOSS Model of Image Generation in Optronic Sensor Systems*, FOA-rapport FOA-R--97-00582-616--SE, 1997
- [9] Carl Hedberg, Lars Tydén, Christer Wigren, *Simulering av målsökare mot störskyddat objekt i en simulerad bakgrund.*, FOI MEMO Dnr:2001-3915.
- [10] Lars Tydén och Lars Berglund, *Laborationshandledning Korrelationsmålföljare*, FOA-rapport FOA-D--97-00335-616--SE, Augusti 1997
- [11] Lars Berglund och Carl Hedberg, *Laborationshandledning: Centroidmålföljare*. FOA-rapport FOA-D--97-00335-616--SE, 1997
- [12] Gustaf Olsson, *Simulering retikelmålsökare - metodbeskrivning*. FOA-rapport FOA C 30685-8.3, 1992.
- [13] <http://www.matrox.com>