

## ارائه معماری بهبودیافته برای درون‌یابی اصلاح مهاجرت برد در الگوریتم رنج داپلر

بهروز ابراهیمی<sup>۱\*</sup>، بیژن وثوقی وحدت<sup>۲</sup>، محمدمهدی نایی<sup>۳</sup>، سعیدرضا محسنی<sup>۴</sup>

۱- کارشناس ارشد ۲- استادیار ۳- استاذ ۴- کارشناس ارشد، دانشگاه صنعتی شریف

(دریافت: ۱۳۹۲/۱۲/۲، پذیرش: ۱۳۹۳/۰۲/۱۳)

### چکیده

یکی از قسمت‌های بسیار مهم در الگوریتم رنج داپلر، اصلاح مهاجرت برد است. در این مرحله برای تخمین داده‌ها در محل واقعی خود از درون‌یابی استفاده می‌شود. دقت درون‌یابی تاثیر بسیار زیادی در کیفیت تصویر نهایی دارد. در مقاله حاضر به بحث درون‌یابی پرداخته شده و اینکه چگونه می‌توان به بهترین تخمین و دقت برای داده‌های مرحله اصلاح مهاجرت برد رسید. برای این منظور، عملکرد فیلترهای مختلف در درون‌یابی از جهت سطح لوب کناری، حد تفکیک در راستای سمت و بار پردازشی، با کمک شبیه‌سازی در یک سناریوی خاص رادار و هدف، مورد بررسی قرار گرفته و از بین آنها فیلتر اسپلاین به عنوان فیلتر برتر برای درون‌یابی برگزیده شده است. اما مشکلی که این فیلتر نسبت به فیلترهای ساده‌تری همچون فیلتر خطی دارد این است که، پیاده‌سازی آن در سخت‌افزار همراه با مشکلاتی از جمله زمان پردازش بالا است. در ادامه مقاله به بررسی معادلات فیلتر اسپلاین پرداخته شده و معماری بسیار ساده خط لوله‌ای برای این فیلتر ارائه شده است. این معماری علاوه بر سادگی آن از جهت پیاده‌سازی، سرعت بالایی نسبت به انواع بازگشتی خود دارد.

### واژگان کلیدی

اصلاح مهاجرت برد، درون‌یابی، پیاده‌سازی، رادار دهانه ترکیبی

### ۱. مقدمه

الگوریتمی است که بتواند پس از پردازش داده‌ها اهداف را در هر دو جهت از هم تفکیک کند [۱-۳]. از جمله مهمترین الگوریتم‌هایی که برای تشکیل تصویر استفاده می‌شود می‌توان به  $CSA^2$ ,  $RDA^2$  و  $\omega k^4$  اشاره کرد [۱، ۴]. یکی از مهم‌ترین و پرکاربردترین این الگوریتم‌ها، الگوریتم رنج داپلر است. الگوریتم رنج داپلر در سال‌های ۱۹۷۸-۱۹۷۶ برای بررسی بر روی داده‌های ماهواره سی‌ست<sup>۵</sup> به کار برده شد و اولین تصویر دیجیتالی SAR مربوط به ماهواره با استفاده از این الگوریتم تولید شد [۵]. در این الگوریتم پس از تشکیل ماتریس دو بعدی سمت و برد برای تفکیک اهداف در راستای برد، از فشرده‌سازی استفاده می‌شود. برای این منظور، داده‌های یک سمت خاص با عبور از فیلتر منطبق فشرده شده و اهداف در راستای برد از هم تفکیک می‌شوند.

رادارهای دهانه ترکیبی (SAR)<sup>۱</sup> با مشخصه وضوح بالا به‌عنوان ابزاری کارآمد برای استخراج تصویر در کاربردهای تجاری و نظامی است. در این رادارها برای داشتن تصویری با کیفیت مناسب باید در هر دو جهت برد و سمت، اهداف را با دقت مناسبی از هم تفکیک کرد. برای تفکیک اهداف در راستای برد با توجه به نوع سیگنالی که استفاده می‌شود، از روش فشرده‌سازی برد استفاده می‌شود. همچنین با توجه به نوع حرکت هواپیما و تصویربرداری که معمولاً به شکل نواری است، با افزایش طول مؤثر دهانه، قدرت تفکیک در راستای سمت بالا می‌رود [۱]. با توجه به پیوستگی اطلاعات سمت و برد در داده SAR نمی‌توان پردازش مستقلی روی داده‌های یک پالس با یک برد خاص انجام داد. به عبارت دیگر، پردازش SAR یک پردازش دو بعدی است. با توجه به این موضوع برای استخراج تصویر، نیاز به

<sup>2</sup> Range Doppler Algorithm

<sup>3</sup> Chirp Scaling Algorithm

<sup>4</sup> Omega K

<sup>5</sup> Seasat

\* رایانامه نویسنده پاسخگو: behrouz\_abraimi67@yahoo.com

<sup>1</sup> Synthetic Aperture Radar (SAR)

مهندسی هستند. غیر از این فیلترها، فیلترهای دیگری همچون درون‌یابی سینک<sup>۴</sup> وجود دارد [۷، ۸]. این فیلترها به دلیل اینکه بار پردازشی بسیار بالایی دارند، مورد بررسی قرار نگرفتند. یکی از اهداف اصلی این مقاله، ارائه معماری به نحوی است که اولاً قابل پیاده‌سازی در سخت‌افزار باشد، ثانیاً نیاز به زمان پردازشی کمتری داشته باشد. در صورت محقق شدن این دو هدف، الگوریتم رنج-دابلر (RDA) را می‌توان به شکل زمان حقیقی پیاده کرد. از بین فیلترهای مورد بررسی، فیلتر اسپلین<sup>۵</sup> در درون‌یابی بهترین عملکرد را در بین فیلترهای مختلف دارد و سطح لوب کناری<sup>۶</sup> آن به شکل قابل ملاحظه‌ای کمتر از فیلترهای دیگر است. این فیلتر گرچه دارای مزایای زیادی از جهت صحت درون‌یابی است، اما با توجه به ساختار آن، پیاده‌سازی این فیلتر به دلیل رابطه بازگشتی، به شکل کاملاً پیچیده است و زمان پردازشی زیادی را می‌طلبد. در مقاله حاضر به این معضل پرداخته شده و معماری فیلتر اسپلین به شکل بهبودیافته‌ای ارائه می‌شود. در این معماری ضمن اینکه بخشی از آن به شکل خط لوله‌ای است و باعث افزایش سرعت می‌شود، از دقت خیلی بالایی هم برخوردار است. با توجه به اینکه در معماری خط لوله‌ای قابلیت رجیستر کردن به راحتی میسر است، از این رو در این معماری‌ها مسیر بحرانی کوچکتر است. این کوچکتر بودن مسیر بحرانی<sup>۷</sup> منجر به افزایش سرعت اجرای الگوریتم در سخت‌افزار می‌شود. در بخش دوم به معرفی کامل فیلتر اسپلین و معماری بهبودیافته آن پرداخته می‌شود. در بخش سوم نتایج حاصل از شبیه‌سازی برای فیلترهای مختلف ارائه می‌شود و سرانجام نتیجه‌گیری انجام می‌شود.

## ۲. معماری اسپلین

فیلتر اسپلین دارای نقش مهمی در قسمت‌های مختلف مهندسی است، که عمدتاً به دلیل ویژگی‌های مطلوب درون‌یابی و ساختار تکه‌ای چندجمله‌ای درون‌یابی آن است [۸-۱۰]. فرمول اسپلین براساس ساختار B-spline قبلاً توسط شونبرگ<sup>۸</sup> در سال ۱۹۴۶ معرفی شده بود و در ادامه با مفهوم تئوری تقریب توسعه داده شد [۱۱]. این ساختار پس از ۳ مقاله [۱۲-۱۴] که به معرفی فیلترینگ براساس رویکرد محاسبه B-spline پرداخته بودند، در انجمن‌های پردازش سیگنال شهرت بسیاری یافت. برتری قاطع فرمولاسیون B-spline بخاطر الگوریتم‌های بهینه آن است. در ادامه، فرمول اسپلین مکعبی براساس ساختار B-spline ارائه می‌شود.

پس از این مرحله، نوبت به تفکیک اهداف در راستای سمت می‌رسد. برای این منظور، قبل از اعمال پردازش بعدی با FFT گرفتن از داده‌های یک برد خاص، داده‌های راستای سمت به حوزه فرکانس می‌روند. مرحله بعد که یکی از مهم‌ترین قسمت‌های الگوریتم رنج-دابلر (RDA) است، باید اصلاح مهاجرت برد (RCMC)<sup>۱</sup> صورت گیرد، تا داده‌های مربوط به یک هدف خاص در یک راستا قرار گیرند. این مرحله یکی از مهم‌ترین مراحل الگوریتم رنج-دابلر (RDA) است، زیرا پس از این مرحله، اهداف در راستای سمت از هم تفکیک می‌شوند. در حقیقت کارایی عملیات اصلاح مهاجرت برد تاثیر بسیار زیادی در کیفیت تصویر نهایی دارد [۶]. در حالت عادی، صفحه داده خام SAR (سمت-برد) در حوزه زمان شامل اطلاعات هدف در طول مرور آنتن است. براساس فیزیک مسئله و بسته به زاویه لوجی<sup>۲</sup>، فاصله مستقیم هدف نسبت به رادار در سمت‌های متوالی متغیر است و به صورت یک منحنی تغییر می‌کند. این منحنی‌ها که همان منحنی‌های مهاجرت برد (RCM)<sup>۳</sup> است، تاریخچه مربوط به یک هدف است، که باید انحنای آن صاف گردد تا در مرحله بعد فشرده‌سازی روی آن صورت گیرد. برای صاف کردن این انحنای داده‌های مربوط به یک برد، از ماتریس داده استخراج‌شده و در یک ماتریس جدید ریخته می‌شود. در نقاطی که منحنی با مرکز سلول فاصله منطبق است، داده خروجی منحنی مهاجرت برد دقیقاً همان داده سلول مورد نظر است. در سمت‌هایی که منحنی در بین دو سلول فاصله از ماتریس داده قرار دارد، باید از درون‌یابی برای خواندن مقدار دقیق داده و ثبت در ماتریس جدید استفاده شود، تا در نهایت با شیفت داده‌های راستای برد، منحنی مهاجرت برد در یک راستا قرار گیرد.

انتخاب درون‌یابی مناسب تاثیر بسیار زیادی در اصلاح مهاجرت برد و کیفیت تصویر نهایی دارد؛ زیرا در صورتی که درون‌یابی به شکل درستی انجام نگیرد، اثر سلول‌های مجاور بر روی محل فعلی خواهد افتاد و محل فعلی مقدار واقعی خود را نخواهد داشت. به این ترتیب در مرحله بعد فشرده‌سازی بر روی داده‌های غیر واقعی انجام می‌شود، که این خود منجر به از دست رفتن کیفیت تصویر نهایی می‌شود. در بررسی‌های انجام‌شده که در قسمت شبیه‌سازی این مقاله آورده شده، فیلترهای مختلفی مورد بررسی قرار گرفته‌اند. این فیلترها عموماً جزء فیلترهای پرکاربرد در شاخه‌های مختلف

<sup>5</sup> Spline

<sup>6</sup> Side Lobe Level (SLL)

<sup>7</sup> Critical Path

<sup>8</sup> Schoenberg

<sup>1</sup> Range Cell Migration Compensation (RCMC)

<sup>2</sup> Squint

<sup>3</sup> Range Cell Migration (RCM)

<sup>4</sup> Sinc Interpolation

این مقدار می‌توان ضرایب  $a_B, b_B, c_B, d_B$  را ساخت و با توجه به مقدار  $\Delta x$  و رابطه (۴) مقدار دقیق تابع در نقطه مورد نظر را درون‌یابی کرد. با توجه به رابطه (۱) برای محاسبه  $W(z)$  نیاز به محاسبه تابع تبدیل  $H(z)$  است. طبق [۱۴] این تابع تبدیل را می‌توان به دو فیلتر پاسخ ضربه نامتناهی (IIR) درجه اول که به شکل سری<sup>۲</sup> هستند، تجزیه کرد.

$$H(z) = \frac{1}{A(z)} = \frac{1}{z+4+z^{-1}} = H_f(z) \cdot H_b(z) \quad (۶)$$

$$H_f(z) = \frac{1}{1-\alpha z^{-1}}, \quad H_b(z) = \frac{z^{-1}}{1-\alpha^{-1}z^{-1}}$$

در این عبارت،  $H_f(z)$  یک تابع علی<sup>۳</sup> است، در صورتی که  $H_b(z)$  تابع تبدیل غیرعلی است. برای تبدیل  $H_b(z)$  به یک تابع علی می‌توان پاسخ ضربه را به یک مقدار مشخص کوتاه<sup>۴</sup> کرد [۸]. بنابراین با این کار ضمن علی کردن سیستم می‌توان از یک فیلتر پاسخ ضربه متناهی (FIR) نیز بهره برد، که این به نوبه خود پیچیدگی پیاده‌سازی فیلتر را به شکل چشمگیری کاهش می‌دهد.

در رابطه (۷)  $\tilde{h}_b[n]$  پاسخ ضربه کوتاه شده و  $\tilde{h}_b(z)$  تبدیل  $z$  آن است.

$$\tilde{h}_b[n] = -\sum_{k=1}^{M+1} \alpha^k \cdot \delta[n-1+k], \quad \tilde{H}_b(z) = -z^{-1} \sum_{k=1}^{M+1} (\alpha z)^k \Rightarrow \quad (۷)$$

$$= \frac{z^{-1}}{1-\alpha^{-1}z^{-1}} \cdot (1-(\alpha z)^{M+1})$$

همانطور که مشاهده می‌شود جمله اول  $\tilde{H}_b(z)$  برابر با مقدار ایده‌آل  $H_b(z)$  است. بنابراین:

$$\tilde{H}_b(z) = H_b(z) \cdot (1-(\alpha z)^{M+1}) \quad (۸)$$

تابع تبدیل کلی عبارت خواهد بود از:

$$\tilde{H}(z) = H_f(z) \cdot \tilde{H}_b(z) = H(z) - (\alpha z)^{M+1} H(z) \quad (۹)$$

آنچه در رابطه (۹) اهمیت دارد، بحث قسمت خطای آن است، که به شکل نمایی وابسته به  $\alpha$  است و با توجه به مقدار  $|\alpha| \cong 0.268$ ، افزایش  $M$  مقدار خطا به سمت صفر میل می‌کند. بنابراین برای افزایش دقت محاسبات می‌توان مقدار  $M$  را افزایش داد، البته برای تعیین مقدار  $M$ ، مصالحه‌ای بین افزایش دقت و میزان

ضرایب B-spline که همان  $w[n]$  است، با فیلتر کردن نمونه‌های ورودی  $y[n]$  یافت می‌شود و تابع تبدیل آن عبارت است از  $H(z) = \frac{1}{A(z)}$  [۱۳]. رابطه (۱) تبدیل  $z$  خروجی فیلتر B-spline را نشان می‌دهد.

$$W(z) = \frac{Y(z)}{A(z)} \quad (۱)$$

برای فرم اسپلاین مکعبی  $A(z)$  فرم متقارن زیر را دارد [۱۳]:

$$A(z) = z + 4 + z^{-1} \quad (۲)$$

به منظور ساخت چند جمله‌ای اسپلاین از روی ضرایب B-spline اجزای بین دو نمونه  $y[n]$  و  $y[n+1]$  باید به شکل جمع ۴ تابع شیفت یافته وزن دار محاسبه شود [۱۳، ۱۵].

$$\beta^3(x) = \begin{cases} 4-3x^2(2-|x|), & x < 0 \\ (2-|x|)^3, & x \geq 0 \\ 0, & \text{others} \end{cases} \quad (۳)$$

$$\bar{Y}_n(x) = \sum_{i=-1}^2 w[n+i] \cdot \beta^3(x-(n-i))$$

رابطه (۳) بیانگر تخمین مقدار نمونه‌های ورودی در محل  $x$  است و با  $y[n]$  طبق رابطه (۴) ارتباط دارد.  $x$  در بازه‌ای بین  $n$  و  $n+1$  می‌تواند تغییر کند و  $x \in R$  است. در تعریف دقیقی که از B-spline مکعبی در [۱۳] شده است، یک ضریب  $\frac{1}{6}$  هم در کنار  $A(z)$  و  $\beta^3(x)$  وجود دارد، که برای سادگی نگارش حذف شده است. این ضریب در کلیت فرمول نقشی ندارد و خللی در درون‌یابی ایجاد نمی‌کند، فقط مقیاس<sup>۱</sup> را تغییر می‌دهد و به عنوان یک ضریب است.  $\bar{Y}_n(x)$  را می‌توان به شکل دیگری نیز بیان کرد که عبارت است از [۸]:

$$\begin{cases} Y_n(\Delta x) = \bar{Y}_n(n+\Delta x) = d_B[n]\Delta x^3 + c_B[n]\Delta x^2 \\ + b_B[n]\Delta x + a_B[n] \\ \Delta x = x - n, \quad 0 \leq \Delta x \leq 1 \end{cases} \quad (۴)$$

که در آن، ضرایب  $a_B, b_B, c_B, d_B$  عبارتست از [۸]:

$$\begin{bmatrix} d_B \\ c_B \\ b_B \\ a_B \end{bmatrix} = \begin{bmatrix} -1 & 3 & -3 & 1 \\ 3 & -6 & 3 & 0 \\ -3 & 0 & 3 & 0 \\ 1 & 4 & 1 & 0 \end{bmatrix} \begin{bmatrix} w[n-1] \\ w[n] \\ w[n+1] \\ w[n+2] \end{bmatrix} \quad (۵)$$

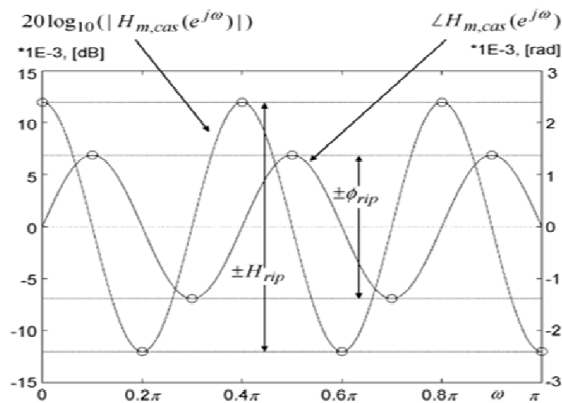
رابطه (۴) عبارت است از چند جمله‌ای b-spline مکعبی، کافی است برای محاسبه مقدار دقیق درون‌یابی، مقدار  $w[n]$  یافت شود. از روی

<sup>۲</sup> Cascade

<sup>۳</sup> Causal

<sup>۴</sup> Truncate

<sup>۱</sup> Scale



شکل ۱. اندازه و فاز تابع تبدیل به ازای  $M=4$  [۸]

در فاز و یا هر دو خطا که کمتر از مقدار ماکزیمم است. با توجه به این موضوع، مقدار خطا در اندازه و فاز فراتر از  $H_{rip}$  و  $\phi_{rip}$  نمی‌رود. جدول ۱ مقادیر ماکزیمم اندازه و فاز ریبیل‌های  $H_{rip}$  و  $\phi_{rip}$  را به ازای مقادیر مختلف  $M$  نشان می‌دهد.

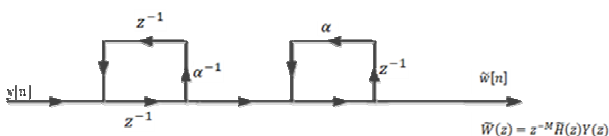
جدول ۱. مقادیر مختلف ماکزیمم خطا به ازای  $M$  های مختلف [۸]

M	۲	۴	۸	۱۶	۳۲
$H_{rip}$ [dB]	$1/7 \times 10^{-1}$	$2/1 \times 10^{-1}$	$2/6 \times 10^{-1}$	$6/1 \times 10^{-1}$	$2/6 \times 10^{-1}$
$\phi_{rip}$ [dB]	$9/1 \times 10^{-2}$	$4/1 \times 10^{-2}$	$1/7 \times 10^{-2}$	$9/1 \times 10^{-2}$	$3/1 \times 10^{-2}$

همانطور که در جدول ۱ مشاهده می‌شود با افزایش  $M$  مقدار خطا کم می‌شود. این خود گویای آن است، که تابع تبدیل به مقدار حقیقی خود نزدیک و درون‌یابی با دقت بالاتری انجام می‌شود.

## ۲-۲. پیاده‌سازی فیلتر:

همانطور که بیان شد فیلتر اسپلاین ساختار تمام بازگشتی دارد. معماری این فیلتر مطابق شکل ۲ است.



شکل ۲. معماری بازگشتی فیلتر اسپلاین

یکی از حسن‌های این ساختار، استفاده از منابع سخت‌افزاری پایین است. معمولاً این معماری در سیستم‌هایی که زمان پردازشی محدود نیست، کاربرد بیشتری دارد. اما ایراد بزرگ این معماری این است که، نیاز به زمان زیادی برای پردازش دارد. در این معماری

سخت‌افزار اشغالی وجود دارد. در ادامه برای رسیدن به علیت باید ضرایب B-spline را به اندازه  $M$  نمونه شیفت دهیم که در نهایت به رابطه (۱۰) می‌رسیم.

$$\tilde{W}(z) = z^{-M} \tilde{H}(z) Y(z) \quad \underline{z}^{-1} \tilde{w}[n+M] = w[n] - e w_M[n] \quad (10)$$

که در آن تبدیل  $z$  خطای ضرایب عبارت‌است از:

$$E W_M(z) = (\alpha z)^{M+1} H(z) Y(z) = (\alpha z)^{M+1} W(z) \quad (11)$$

## ۲-۱. محاسبه خطا:

فیلتر B-spline به واسطه کوتاه شدن، خطایی در سیستم ایجاد می‌کند. این خطا در تعیین مقدار  $M$  اهمیت پیدا می‌کند، از این رو برای محاسبه مقدار خطا داریم:

$$\tilde{H}(z) = H(z) H_{M,cas}(z), \quad H_{M,cas}(z) = 1 - (\alpha z)^{M+1} \quad (12)$$

مقدار اندازه خطا طبق رابطه (۱۳) به دست می‌آید و این مقدار در محدوده  $1 \pm |\alpha|^{M+1}$  قرار می‌گیرد [۸].

$$|H_{M,cas}(e^{j\omega})|^2 = 1 - 2\alpha^{M+1} \cos(\omega(M+1)) + \alpha^{2(M+1)} \quad (13)$$

$$1 - |\alpha|^{M+1} \leq |H_{M,cas}(e^{j\omega})| \leq 1 + |\alpha|^{M+1}$$

از آنجایی  $|\alpha|^{M+1}$  که همیشه کمتر از یک است، بنابراین نوسانات می‌تواند به‌طور تقریبی در محدوده زیر قرار گیرد [۸].

$$\left| 20 \log_{10} \left( |H_{M,cas}(e^{j\omega})| \right) \right| < H_{rip} \cong \frac{20}{\ln(10)} |\alpha|^{M+1} \quad (14)$$

فاز تابع تبدیل نیز به شکل زیر است:

$$\angle H_{M,cas}(e^{j\omega}) = -\text{atan} \frac{\alpha^{M+1} \sin(\omega(M+1))}{1 - \alpha^{M+1} \cos(\omega(M+1))} \quad (15)$$

فاز تابع تبدیل در حوالی صفر نوسان می‌کند و ریبیل آن طبق رابطه (۱۶) به دست می‌آید [۸].

$$\left| \angle H_{M,cas}(e^{j\omega}) \right| \leq \phi_{rip} = \text{atan} \frac{|\alpha|^{M+1}}{\sqrt{1 - \alpha^{2(M+1)}}} \quad (16)$$

اندازه و فاز تابع  $H_{M,cas}$  تبدیل به ازای  $M=4$  در شکل ۱ نشان داده شده است. همانطور که در شکل ۱ مشاهده می‌شود در جاهایی که اندازه خطا به ازای فرکانسی ماکزیمم شده است، در همان فرکانس فاز خطا صفر شده و در فرکانس‌هایی که فاز خطا ماکزیمم شده، اندازه خطا صفر شده است. بنابراین در درون‌یابی براساس فرمول B-spline علی‌بسته به اینکه مقدار فرکانس چقدر باشد، یا دارای ماکزیمم خطا در اندازه است، یا دارای ماکزیمم خطا

جدول ۲. مقدار منابع مورد استفاده در معماری ارائه شده

تاخیر دهنده	ضرب کننده	جمع کننده	
M	M	M-1	$\tilde{H}_{bm}(z)$
1	1	1	$H_f(z)$
M+1	M+1	M	Total

همانطور که مشاهده می شود شکل ۴ پیاده کننده رابطه (۱۰) است. با استفاده از این معماری می توان فیلتر اسپلاین را در درون یابی به کار برد و اصلاح مهاجرت برد را با دقت بالایی در سیستم اعمال کرد. جدول ۲ نشان دهنده میزان منابع استفاده شده برای این پیاده سازی است.

در [۱۶-۱۸] نویسندگان روشی پیشنهاد داده اند که تماماً به شکل پاسخ ضربه متناهی (FIR) پیاده شده است. این سبک معماری، منابع سخت افزاری بالایی نسبت به معماری حاضر می برد، زیرا در ساختارهایی که تماماً به شکل پاسخ ضربه متناهی (FIR) تقریب زده می شود، از تعداد ضرب کننده، جمع کننده و تاخیر دهنده بیشتری استفاده می شود. از طرفی هم چون تقریب بیشتری در آن به کار رفته، جواب همراه با خطای بیشتری وجود دارد. در معماری حاضر، مصالحه ای بین میزان دقت در سیستم و حجم سخت افزاری اشغالی وجود دارد. اما عیبی که در معماری ارائه شده وجود دارد این است که، چون بخشی از آن به شکل پاسخ ضربه نامتناهی (IIR) است، برای آن، ساختار تمام خط لوله ای نمی توان متصور شد، در صورتی که در ساختارهایی که به صورت تماماً پاسخ ضربه متناهی (FIR) است این امر ممکن است. معماری ارائه شده در واقع مصالحه ای بین معماری تمام بازگشتی و معماری تمام خط لوله ای است

ایده اصلی در واقع استفاده از فیلتر اسپلاین به عنوان بهترین درون یاب برای بخش اصلاح مهاجرت برد و نیز ارائه معماری است، که به سادگی و به شکل بهبود یافته ای در سخت افزار قابل پیاده سازی باشد و در ضمن از دقت خوبی نیز برخوردار باشد.

### ۳. شبیه سازی

برای آنکه بتوان روش های مختلف درون یابی را مقایسه کرد و بهترین روش را استخراج کرد، یک هدف نقطه ای مطابق با پارامترهای جدول ۳ شبیه سازی شده است. برای این شبیه سازی از هندسه رادار و هدف مطابق با شکل ۵ استفاده شده است.

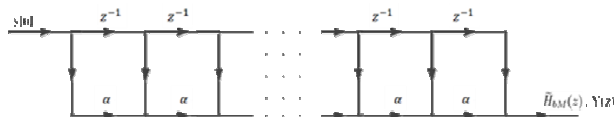
تا زمانی که پردازش قسمت اول تمام نشود، پردازش قسمت دوم بی تأثیر است. از این رو برای پردازش داده ها نیاز به مدت زمان  $t_1+t_2$  ثانیه است.  $t_1$  مدت زمان پردازش قسمت اول و  $t_2$  مدت زمان پردازش قسمت دوم معماری است. برای بهبود معماری شکل ۲ و کاهش زمان پردازش از رابطه (۷) کمک گرفته شده است. برای پیاده سازی خط لوله ای قسمت اول فیلتر از  $\tilde{H}_{bm}(z)$  استفاده می شود.

$$\tilde{H}_{bm}(z) = z^{-M} \tilde{H}_b(z) = \sum_{k=0}^M \alpha^{M+k} z^{-k} = -(\alpha^{M+1} + \alpha^M z^{-1} + \dots + \alpha^2 z^{-(M-1)} + \alpha z^{-M}) \quad (17)$$

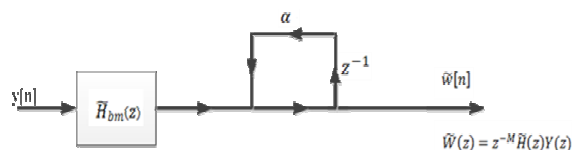
رابطه (۱۷) یک فیلتر پاسخ ضربه متناهی (FIR) است، که معماری آن مطابق شکل ۳ است. در این معماری که تشکیل شده است از مجموعه ای از ضرب کننده ها، جمع کننده ها و تاخیر دهنده ها داده ها به شکل خط لوله ای به آن وارد و از آن خارج می شوند. اصولاً یکی از مزیت های اصلی فیلترهای پاسخ ضربه متناهی (FIR)، قابلیت خط لوله ای کردن آنهاست. به همین دلیل، در معماری هایی که ارائه می شود، معمولاً از این سبک فیلترها استفاده می شود، تا سرعت سیستم را افزایش دهند.

یکی از ویژگی های مهم در معماری شکل ۳ بحث خط لوله ای بودن آن است. این ویژگی سبب می شود که داده ها به شکل سیلابی به آن وارد و از آن خارج شوند. این امر سبب می شود که مدت زمان پردازشی بسیار کاهش یابد و از این جهت سرعت سیستم افزایش یابد. با استفاده از معماری شکل ۳ فیلتر اسپلاین را تکمیل می کنیم.

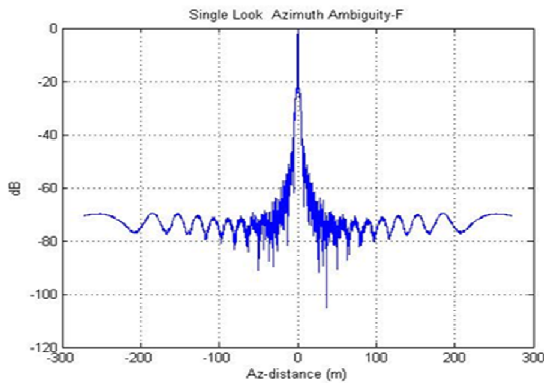
در معماری شکل ۴ مدت زمان پردازش به زمان  $t_2$  ثانیه کاهش یافت، زیرا قسمت اول به شکل خط لوله ای پیاده شده است و داده ها به شکل سیلابی به آن وارد می شود. تنها قسمت دوم که به شکل بازگشتی است، نیاز به زمانی برای پردازش دارد. مزیت اصلی معماری شکل ۴ نسبت به شکل ۲ مدت زمان پردازش آن است. در معماری بهبود یافته ما به اندازه  $t_1$  ثانیه صرفه جویی می کنیم.



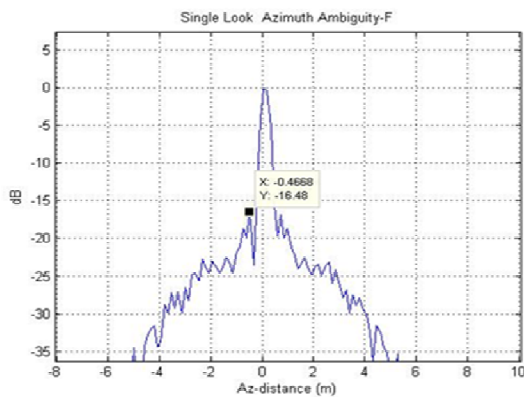
شکل ۳. معماری پاسخ ضربه متناهی (FIR) فیلتر



شکل ۴. معماری فیلتر اسپلاین



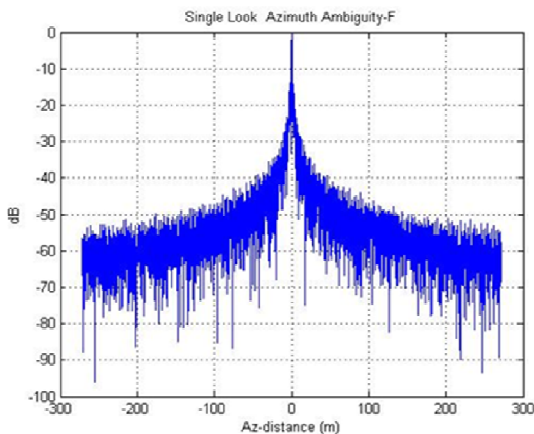
شکل ۶. نمای کلی تابع ابهام به ازای اعمال فیلتر خطی



شکل ۷. نمای نزدیک تابع ابهام به ازای اعمال فیلتر خطی

### ۳-۲. درون‌یابی با فیلتر Nearest Neighbor

در ادامه به جای فیلتر خطی، از فیلتر nearest neighbor استفاده شده است. با اعمال فیلتر nearest neighbor در این درون‌یابی میزان سطح لوب کناری (SLL) در تابع ابهام برابر ۱۴dB شده است. حد تفکیک در راستای سمت نیز به ازای اعمال این فیلتر برابر ۰/۲m است.

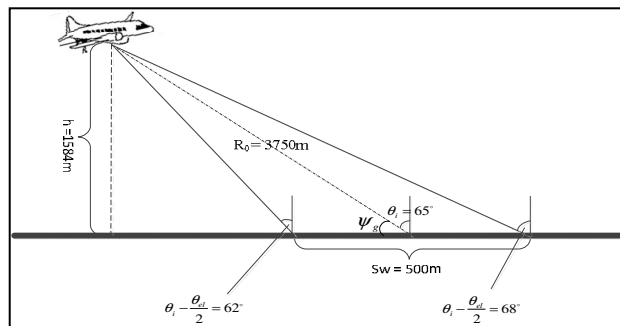


شکل ۸. نمای کلی تابع ابهام به ازای اعمال فیلتر Nearest Neighbor

تابع ابهام این هدف نقطه‌ای به ازای فیلترهای مختلف بررسی شده است. در نهایت با توجه به سطح لوب کناری (SLL)، حد تفکیک و بار پردازشی، بهترین درون‌یابی انتخاب شده است.

### جدول ۲. مقدار منابع مورد استفاده در معماری ارائه شده

طول دهانه ترکیبی (m)	فاصله از هدف (m)	آنتن پهنای بزم (degree)	سرعت حرکت هواپیما (m/s)	نرخ ارسال پالس (Hz)
۳۹۲/۶	۳۷۵۰	۶	۵۰	۳۰۰۰



شکل ۵. هندسه رادار و هدف

داده‌های الگوریتم رنج-داپلر (RDA) یک ماتریس دو بعدی برد-داپلر است. این ماتریس پس از انجام عملیات اصلاح مهاجرت برد، در راستای سمت فشرده‌سازی می‌شود و خروجی آن، تابع ابهام راستای سمت را می‌سازد. این تابع ابهام طبق رابطه (۱۸) محاسبه می‌شود [۱].

$$S = 10 \log_{10} |A_0 p_r(\tau - \frac{2R_0}{c}) p_a(\eta)| \times \exp(\frac{-j4\pi f_0 R_0}{c}) \times \exp(j2\pi f_0 \eta) \quad (18)$$

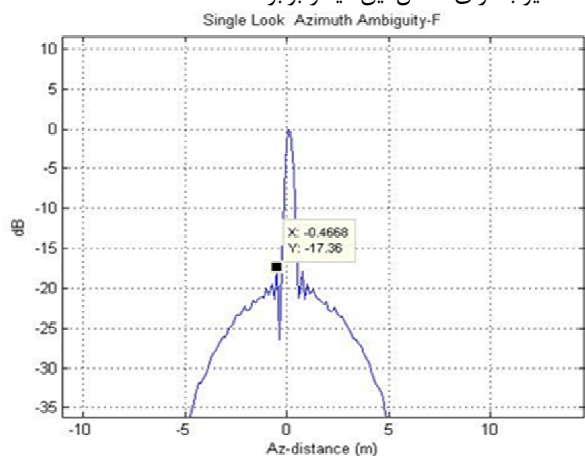
در رابطه (۱۸)  $\eta$  زمان راستای سمت،  $\tau$  زمان راستای برد و  $p_a$  دامنه پاسخ ضربه در راستای سمت است.

### ۳-۱. درون‌یابی خطی

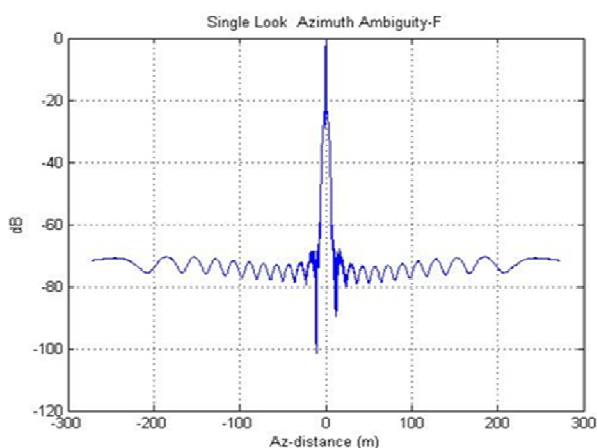
در این شبیه‌سازی، در قسمت اصلاح مهاجرت برد به جای فیلتر مورد نظر برای درون‌یابی، از فیلتر خطی استفاده شده است. همانطور که در شکل مشخص است، میزان سطح لوب کناری (SLL) در تابع ابهام، به ازای اعمال درون‌یابی با فیلتر خطی برابر ۱۶/۴۸dB است. یکی از مزیت‌های اصلی فیلتر خطی، سادگی پیاده‌سازی آن است، که می‌تواند گزینه مناسبی برای پیاده‌سازی در درون‌یابی سیستم باشد. حد تفکیک در راستای سمت نیز به ازای اعمال این فیلتر برابر ۰/۲۲m است.

<sup>1</sup> Pulse Repetition Frequency

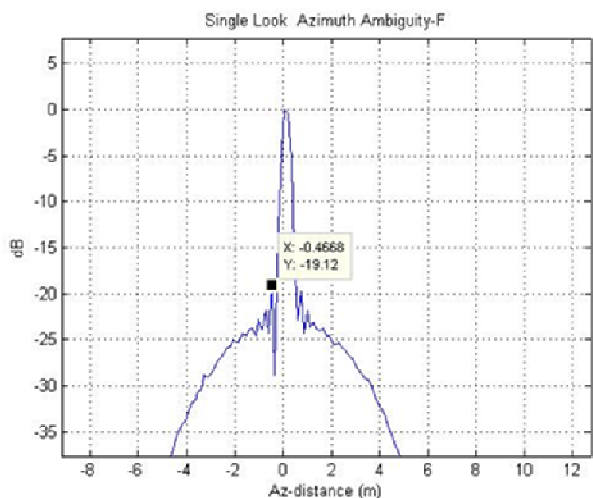
وضعیت این فیلتر نسبت به بقیه فیلترها بهتر است، اما همچنان مشکل پیچیدگی پیاده‌سازی وجود دارد. حد تفکیک در راستای سمت نیز به ازای اعمال این فیلتر برابر  $0.17\text{m}$  است.



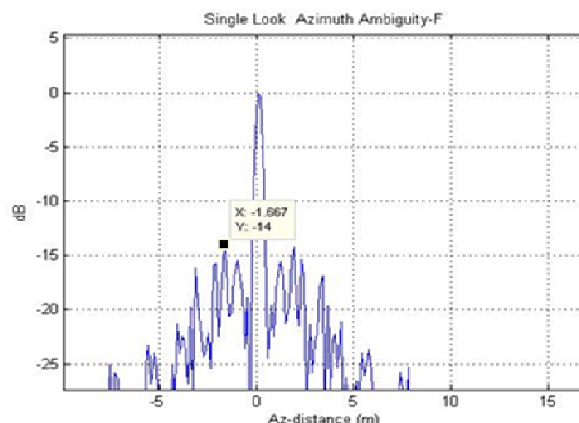
شکل ۱۰. نمای نزدیک تابع ابهام به ازای اعمال فیلتر Piecewise Cubic



شکل ۱۱. نمای کلی تابع ابهام به ازای اعمال فیلتر اسپلاین



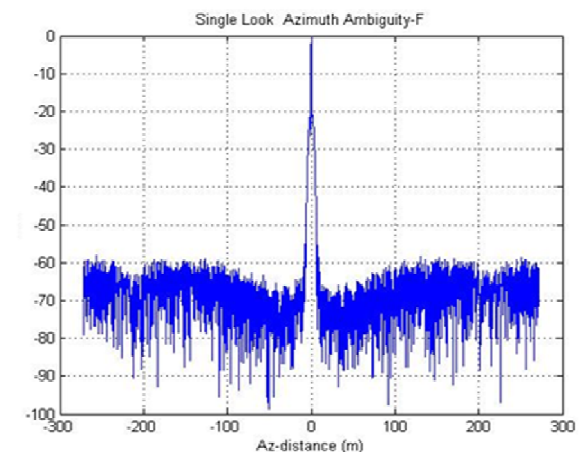
شکل ۱۲. نمای نزدیک تابع ابهام به ازای اعمال فیلتر اسپلاین



شکل ۹. نمای نزدیک تابع ابهام به ازای اعمال فیلتر Nearest Neighbor

### ۳-۳. درون‌یابی با فیلتر Piecewise Cubic Hermite

فیلتر بعدی که به سیستم اعمال شده است، فیلتر Piecewise cubic Hermite یا همان PCHIP است. با استفاده از فیلتر PCHIP مقدار سطح لوب کناری (SLL) در تابع ابهام برابر  $17/36\text{dB}$  شده است. سطح لوب کناری (SLL) در تابع ابهام به ازای این فیلتر وضعیت بهتری نسبت به فیلتر nearest neighbor دارد و وضعیت نسبتاً مشابهی با فیلتر خطی دارد، اما مشکل پیچیدگی از لحاظ سخت‌افزاری همچنان در این فیلتر وجود دارد. حد تفکیک در راستای سمت نیز به ازای اعمال این فیلتر برابر  $0.17\text{m}$  است.



شکل ۱۰. نمای کلی تابع ابهام به ازای اعمال فیلتر Piecewise Cubic

### ۳-۴. درون‌یابی با فیلتر اسپلاین

آخرین فیلتری که به سیستم اعمال شده است فیلتر اسپلاین است. با استفاده از فیلتر اسپلاین مقدار سطح لوب کناری (SLL) برابر  $19/12\text{dB}$  شده است. میزان سطح لوب کناری (SLL) در تابع ابهام به‌ازای این فیلتر به مقدار چشم‌گیری از بقیه فیلترها کمتر است و

پیچیده‌تر است. فیلتر خطی دارای معماری بسیار ساده و قابل پیاده‌سازی است، اما مشکلی که دارد این است که، به نسبت فیلتر اسپلاین دارای سطح لوب کناری (SLL) بالاتری است. این امر ضمن بالا بردن احتمال ابهام، باعث کاهش کیفیت تصویر خروجی نیز می‌شود. پس در صورتی که بتوان مشکل پیچیدگی اسپلاین را حل کرد، می‌تواند به‌عنوان بهترین گزینه برای بخش درون‌یابی اصلاح مهاجرت برد باشد و داده‌ها را با دقت بسیار بالایی تخمین بزند.

#### ۴. نتیجه‌گیری

در این مقاله نشان دادیم که با استفاده از فیلتر اسپلاین می‌توان به بهترین سطح لوب کناری (SLL) در تابع ابهام رسید و درون‌یابی را با دقت بالایی انجام داد؛ به طوری که تصویر نهایی از کیفیت بالایی برخوردار باشد. در ضمن بیان شد که فیلتر اسپلاین به دلیل ساختار بازگشتی آن از پیچیدگی‌های زیادی برخوردار است و زمان پردازشی زیادی را برای انجام آن می‌طلبد. از این رو با ارائه معماری پیشنهادی، پیاده‌سازی آن را در سخت‌افزار به شکل ساده‌تری میسر کردیم، در ضمن با ارائه این معماری مصالحه‌ای بین میزان استفاده از سخت‌افزار و دقت اندازه‌گیری برقرار کردیم. با توجه به این نکات می‌توان در سیستم‌های SAR برای پیاده‌سازی الگوریتم رنج داپلر در قسمت اصلاح مهاجرت برد از درون‌یابی با فیلتر اسپلاین استفاده کرد، که ضمن رسیدن به دقت بالاتر و در نتیجه کیفیت تصویر بیشتر، از لحاظ پیاده‌سازی هم بسیار سهل و بهینه باشد.

جدول ۴، مقایسه روش‌های مختلف از جنبه سطح لوب کناری و حد تفکیک راستای سمت است. همانطور که مشاهده می‌شود، از جهت حد تفکیک وضعیت فیلترهای PCHIP و Spline مشابه هم و بهتر از دو فیلتر دیگر است. از جهت سطح لوب کناری نیز دو فیلتر PCHIP و Spline در وضعیت بهتری نسبت به دو فیلتر دیگر قرار دارند، اما در مقایسه بین این دو فیلتر، Spline با ۲dB بهبود بیشتر به دقت بهتری رسیده است. برای مقایسه دقیق‌تر بین فیلترهای بیان شده، مقایسه‌ای هم از جهت بار پردازشی شده است. این مقایسه در قالب جدول ۵ نشان داده شده است.

همانطور که در جدول ۵ نشان داده شده، از لحاظ بار پردازشی فیلترهای Linear و Nearest در وضعیت بهتری قرار دارند و بار پردازشی آن‌ها نسبت به دو فیلتر دیگر کمتر است. بین دو فیلتر PCHIP و Spline نیز فیلتر Spline از تعداد ضرب‌کننده کمتری بهره می‌برد، که در مقایسه بین این دو Spline بار پردازشی کمتری دارد.

با توجه به نتایج به‌دست آمده و بررسی نتیجه فیلترهای مختلف در تابع ابهام، درون‌یابی که از فیلتر اسپلاین به‌دست آمده، دارای کمترین میزان سطح لوب کناری (SLL) است، که این امر باعث کاهش ابهام و افزایش کیفیت تصویر به‌دست آمده می‌شود. از طرفی پیاده‌سازی این فیلتر نسبت به فیلتر خطی بسیار مشکل‌تر و

جدول ۴. مقایسه روش‌های مختلف از جنبه سطح لوب کناری و حد تفکیک سمت

	Linear	Nearest Neighbor	PCHIP	Spline
سطح لوب کناری (db)	-۱۶/۴۸	-۱۴	-۱۷/۳۶	-۱۹/۱۲
حد تفکیک سمت (m)	۰/۲۲	۰/۲	۰/۱۷	۰/۱۷

جدول ۵. مقایسه بین فیلترها از جهت بار پردازشی

	Linear	Nearest Neighbor	PCHIP	Spline
تعداد ضرب	۲	۳	۳۰	۱۳
تعداد جمع	۴	۱	۸	۸
تعداد کل	۶	۴	۳۸	۲۱



## ۵. مراجع

- [9] Meijering, E., "A Chronology of Interpolation: From Ancient Astronomy to Modern Signal and Image Processing," Proceedings of the IEEE, Vol. 90, pp. 319-342, 2002.
- [10] Unser, M., "Splines: A perfect fit for signal and image processing," Signal Processing Magazine, IEEE, Vol. 16, pp. 22-38, 1999.
- [11] de Boor, C., "On calculating with B-spline," J Approx Theor., Vol. 6, No. 1, pp. 50-62, 1972.
- [12] Unser, M., Aldroubi, A., and Eden, M., "Fast B-spline transforms for continuous image representation and interpolation," IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 13, pp. 277-285, 1991.
- [13] Unser, M., Aldroubi, A., and Eden, M., "B-spline signal processing. I. Theory," Signal Processing, IEEE Transactions on, Vol. 41, pp. 821-833, 1993.
- [14] Unser, M., Aldroubi, A., and Eden, M., "B-spline signal processing. II. Efficiency design and applications," Signal Processing, IEEE Transactions on, Vol. 41, pp. 834-848, 1993.
- [15] Schoenberg, I. J., "Contributions to the problem of approximation of equidistant data by analytic functions," Quart. Appl. Math., Vol. 4, pt. A, pp. 45-99, 1946.
- [16] Ming-Lei, L., "Spline Fit Made Easy," Computers, IEEE Transactions on, Vol. C-25, pp. 522-527, 1976.
- [17] Nagahara, M., Wada, T., and Yamamoto, Y., "Causal spline interpolation by  $H^\infty$  optimization," in Proc. 2006 SICE-ICASE Int. Joint Conf., Convention Center-BEXCO, Busan, Korea, pp. 4160-4163, Oct. 2006.
- [18] Nagahara, M., and Yamamoto, Y., "Causal spline interpolation by  $H^\infty$  optimization," in Proc. Int. Conf. Acoustics, Speech, Signal Processing (ICASSP), Vol. 3, pp. III-1469-III-1472, 2007.
- [1] Cumming, I. G. and Wong, F. H. C., Digital Signal Processing of Synthetic Aperture Radar Data: Algorithms and Implementation: Artech House, Incorporated, 2005.
- [2] Cafforio, C., Prati, C., and Rocca, F., "SAR Data Focusing Using seismic Migration Techniques," Aerospace and Electronic Systems, IEEE Transactions on, Vol. 27, pp. 194-207, 1991.
- [3] Raney, R. K., Runge, H., Bamler, R., Cumming, I. G., and Wong, F. H., "Precision SAR processing using chirp scaling," Geoscience and Remote Sensing, IEEE Transactions on, Vol. 32, pp. 786-799, 1994.
- [4] Linyu, W., Yanxia, Z., "An Improved Algorithm of Range-Doppler for Air-borne Synthetic Aperture Radar", International Conference on Transportation, Mechanical, and Electrical Engineering (TMEE) December 16-18, Changchun, China, 2011.
- [5] Bennett, J. R., Cumming, I. G., Deane, R. A., Widmer, P., Fielding, R., and McConnell, P., SEASAT Imagery Shows St. Lawrence. Aviation Week and Space Technology, page 19 and front cover, 1979.
- [6] C. Dai, X. Zhang, and J. Shi, "Range Cell migration Correction for Bistatic SAR Image Formation," Geoscience and Remote Sensing Letters, IEEE, Vol. 9, pp. 124-128, 2012.
- [7] Candocia, F., and Principe, J. C., "Comments on Sinc Interpolation of Discrete Periodic Signals," Signal Processing, IEEE Transactions on, Vol. 46, pp. 2044-2047, 1998.
- [8] Petrinovic, "Causal Cubic Splines: Formulations, Interpolation Properties and Implementations," Signal Processing, IEEE Transactions on, Vol. 56, pp. 5442-5453, 2008.

## **Introducing an Improved Implementation for the Interpolation of Range Cell Migration Correction in Range Doppler Algorithm**

**B. Ebrahimi\*, B. Vosoughi Vahdat, M. M. Nayebi, S. R. Mohseni**

\* Sharif University of Technology

(Received:02/21/2014, Accepted:05/03/2014)

### **Abstract**

One of the most important parts of range Doppler algorithm is range cell migration correction. In this part, an interpolation approach is used to estimate data in their correct location. The interpolation method has a great influence on the quality of ultimate image. In this paper the interpolation method and different ways for increasing its accuracy in range cell migration correction section are discussed. The performance of different filters of the side lobe level, resolution in cross range and processing time are studied and then among all of them the spline filter method is chosen as the best filter for the interpolation. Since the implementation of spline filter needs more processing time rather than linear filter, a very simple pipe line architecture is introduced for this filter which not only needs simple hardware but also increases processing speed.

**Keywords:** Rang Cell Migration Correction, Interpolation, Implementation, Synthetic Aperture Radar.