Article

A Comparison of Processing Schemes for Automotive MIMO SAR Imaging

Marco Manzoni [1,*], Stefano Tebaldini [1], Andrea Virgilio Monti-Guarnieri [1] and Claudio Maria Prati [1] and Ivan Russo [2]

1   Dipartimento di Elettronica Informazione e Bioingegneria (DEIB), Politecnico di Milano, Via Ponzio 34/5, 20133 Milano, Italy
2   Huawei Technologies Italia S.r.l., 20129 Milano, Italy
*   Correspondence: marco.manzoni@polimi.it

Abstract: Synthetic Aperture Radar (SAR) imaging is starting to play an essential role in the automotive industry. Its day and night sensing capability, fine resolution, and high flexibility are key aspects making SAR a very compelling instrument in this field. This paper describes and compares three algorithms used to combine low-resolution images acquired by a Multiple-Input Multiple-Output (MIMO) automotive radar to form an SAR image of the environment. The first is the well-known Fast Factorized Back-Projection (FFBP), which focuses the image in different stages. The second one will be called 3D2D, and it is a simple 3D interpolation used to extract the SAR image from the Range-Angle-Velocity (RAV) data cube. The third will be called Quick&Dirty (Q&D), and it is a fast alternative to the 3D2D scheme that exploits the same intuition. A rigorous mathematical description of each algorithm is derived, and their limits are addressed. We then provide simulated results assessing different interpolation kernels, proving which one performs better. A rough estimation of the number of operations proves that both algorithms can be deployed using a real-time implementation. Finally, we will present some experimental results based on open road campaign data acquired using an eight-channel MIMO radar at 77 GHz, considering the case of a forward-looking geometry.

Keywords: radar; SAR; automotive; focusing; FFBP; car-based; MIMO

1. Introduction

In recent years, automation is becoming more and more critical in the automotive industry. Simple parking assistance systems are becoming standard technology for modern cars, while fully autonomous driving systems appear to be just a few years ahead of us. All these technologies are supported by a variety of different sensors such as optical cameras, LiDAR, and radars [1].

In the literature, several imaging algorithms are detailed using one or the combination of more sensors. In [2], for example, a LiDAR is used to obtain an accurate map of the surrounding environment in combination with a refined scanning algorithms allowing for a much lower scanning time. In [3], instead, a super-resolution algorithm is applied to improve the accuracy of the urban mapping.

This paper, however, treats the problem of imaging using radars. Such sensors have specific peculiarities that make them attractive to the automotive industry. First of all, the radar is an active sensor; therefore, the instrument provides illumination. Being able to control the illumination of the scene allows for flexibility in terms of spatial resolution (that is adjustable by changing the bandwidth of the transmitted signal) and permits day and night imaging of the environment. Radars operating at microwave are also not particularly sensitive to fog, rain, and snow, which are three typical environmental scenarios that create troubles for LiDAR and cameras.

The drawback of conventional automotive radar imaging is the poor angular resolution. Such systems typically employ a Multiple-Input Multiple-Output (MIMO) architecture

to generate a virtual array that achieves some angular resolution. However, such resolution is bounded by the length of the array and could be in the order of tens of degrees [4].

Recently, a lot of effort has been carried out in the field of automotive-based Synthetic Aperture Radar (SAR) imaging. This technique, known for decades in space-borne and air-borne radar imaging, has been rarely applied in the automotive industry. SAR imaging exploits several radar pulses transmitted by a moving platform (in this case, a vehicle) to synthesize a very long array leading to a much finer resolution than conventional radar imaging [5].

The first experiments for what concerns automotive SAR imaging are the ones carried out in [6] using a very high frequency and wide bandwidth radar mounted on a vehicle. This work shows the capabilities of SAR systems but requires costly hardware and it is not suited for real-time applications due to large computational burden.

Recent works also demonstrated the capabilities of car-based SAR imaging to map the urban environment. In [7,8], the authors carried out several laboratory tests using a forward-looking SAR and by exploiting compressive sensing. Other works include the one in [9] where a TDM (Time Division Multiplexing) MIMO radar has been used to form SAR images of the surroundings. The well-known focusing algorithm (the Range–Doppler algorithm with secondary migration correction) was also tested in the automotive scenario showing good performances, at least for linear trajectories [10]. The state of the art is currently represented by [11–13] where real data are used, which were acquired in an urban scenario and exploiting non-linear trajectories of the car.

In [14], the necessity of an autofocus algorithm has been made clear: fine image resolution and short wavelengths demand high accuracy in the knowledge of the vehicle's trajectory. The requested accuracy can be as low as 1–2 cm/s. Such precision cannot be reached by standard automotive grade Navigation Units (NU); thus, a proper autofocusing algorithm is mandatory. The approach proposed in [14] exploits a set of low-resolution images focused thanks to the presence of a virtual array to recover a residual motion velocity, leading to a well-focused and well-localized image.

The contribution of this paper consists of describing and comparing three algorithms that can be used to combine low-resolution MIMO images to obtain the final focused SAR image. The analytical foundation of each algorithm is derived, and their limitations are also explained in details.

In Section 2, the geometry of the system is explained and the processing steps that lead to a set of coregistered MIMO images are detailed.

In Section 3.1, the first algorithm, the Fast-Factorized Back-Projection (FFBP) is described, while in Section 3.2, the same thing is completed on the second algorithm, which is called 3D2D. The origin of this name will be explained in the corresponding section. In the same section, we propose a variation of the algorithm called Quick & Dirty (Q&D). It is a version of 3D2D which avoids the compensation of range migration and phase curvature. Its strong limitations compared to the two previously listed algorithms will be detailed in the same section.

Section 4 provides results with simulated data. This section compares the two algorithms using different interpolation kernels and different aperture lengths by assessing the quality of the Impulse Response Function (IRF) using appropriate metrics such as the normalized peak value and the Integrated Side Lobe Ration (ISLR). In Section 5, a rough estimate of the number of operations required by the algorithms to form the final SAR image is provided to the reader.

Section 6 provides the results of the open road campaign carried out using a 77 GHz MIMO radar in a forward-looking configuration. Finally, in Section 7, the conclusions are drawn.

2. Acquisition Geometry and MIMO Image Formation

2.1. Geometry and Signal Model

The system is composed of a vehicle (the moving platform) and a MIMO radar. The radar transmits an impulse with a specific bandwidth, leading to a slant range resolution of [15]:

$$\rho_r = \frac{c}{2B} \tag{1}$$

where ρ_r is the slant range resolution, c is the speed of light and B is the transmitted bandwidth.

Thanks to the presence of multiple channels, the radar generates the so-called *virtual* array obtaining in this way angular resolution. The angular resolution depends on the wavelength, and it is inversely proportional to the length of the virtual array:

$$\rho_\phi = \frac{\lambda}{2N_{ch}d \cos\left(\phi - \phi_{inst}\right)} \tag{2}$$

where ρ_ϕ is the angular resolution, $\lambda = c/f_0$ is the central wavelength of the radar (f_0 being the central frequency), N_{ch} is the number of channels (Virtual Antenna Phase Center, VAPC), d is the spacing between each virtual antenna, ϕ is the azimuth angle and ϕ_{inst} is the radar installation angle.

In Figure 1a, the geometry of the forward-looking system is depicted ($\phi_{inst} = 0$ deg). For the sake of simplicity, the 2D scenario is represented. The antenna elements are the red dots displaced in the direction orthogonal to the motion.
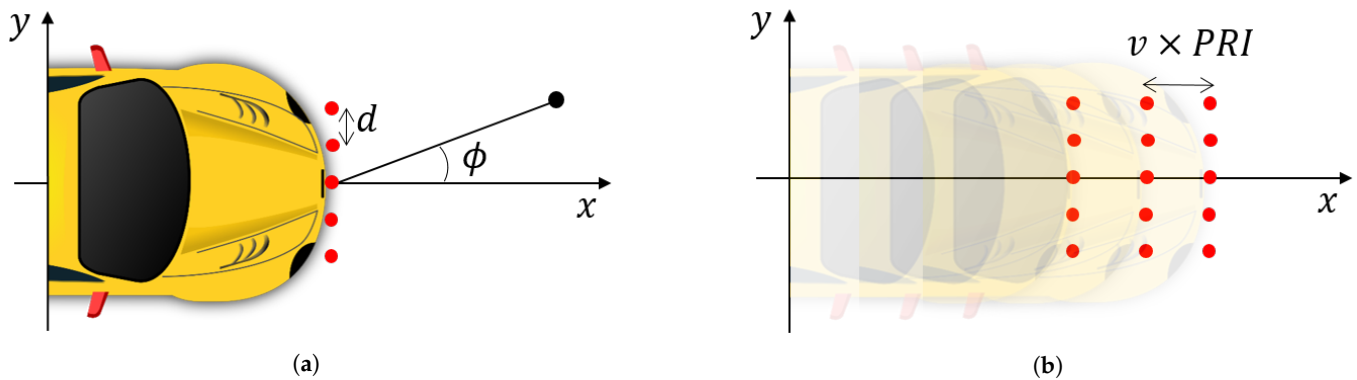

(**a**)

(**b**)

Figure 1. Geometry of the system. The red dots are the virtual channels of the MIMO radar with a distance d, ϕ is the azimuth angle, and the black dot is a generic location is the field of view. In this figure, the installation angle is $\phi_{inst} = 0$ deg, representing a forward-looking geometry (**a,b**).

While the car is moving, the MIMO radar transmits and receives the backscattered echos. We remark that for each time instant τ, a number of signals equal to N_{ch} are received. In Figure 1b, a car travels at velocity v in a straight trajectory and acquires every PRI seconds (Pulse Repetition Interval) a set of $N_{ch} = 5$ signals.

The raw data are compressed [16], leading to the range of compressed data $s_{rc}(r, i; \tau_n)$ where r is the range, i is the index of the sensor ($i = 1, 2, \ldots, N_{ch}$), and τ_n is the sampled slow-time variable. By neglecting the bistatic nature of the MIMO array, we can write the model of the range compressed data for a point target located at polar coordinates (r_0, ϕ_0):

$$s_{rc}(r, i; \tau_n) = \text{sinc}\left[\frac{r - R(r_0, \phi_0, i, \tau_n)}{\rho_r}\right] \exp\left\{-j\frac{4\pi}{\lambda}R(r_0, \phi_0, i, \tau_n)\right\} \tag{3}$$

where sinc is the cardinal sine function, and $R(r_0, \phi_0, i, \tau)$ is the one-way distance from the ith VAPC to the target at the time instant τ_n. Notice that in Equation (3), we have also neglected an amplitude factor due to the spread loss and the Radar Cross-Section (RCS) of the target.

2.2. MIMO Image Formation

The N_{ch} simultaneous acquisitions can be used to focus a low-resolution image by a simple back-projection. In order to achieve high performances, it is convenient to avoid the back-projection on a Cartesian grid: the unnecessary high sampling of the signal leads to an unsustainable computational burden for real-time applications.

A significant improvement in performances can be achieved by back-projecting the signal on a polar grid [17]. In this case, a fixed polar grid is defined in space as in Figure 2 where each pixel has a range (r) and angular position (ϕ) defined with respect to the origin of the grid.
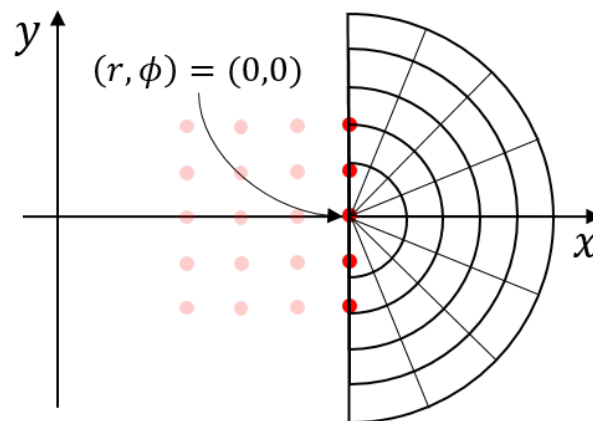

Figure 2. Fixed polar grid where the signal is back-projected. Notice that this grid is strictly polar just for the last acquisition where the origin of the grid is co-located with the center of the virtual array.

The low-resolution images can be formed now by a simple back-projection:

$$I_{MIMO}(r, \phi, \tau) = \sum_{i=1}^{N_{ch}} s_{RC}[R(r, \phi, i, \tau_n), i; \tau_n] \exp\left\{+j\frac{4\pi}{\lambda}R(r, \phi, i, \tau_n)\right\} \qquad (4)$$

where $I_{MIMO}(r, \phi, \tau_n)$ is the low-resolution radar image at coordinates r and ϕ, acquired at slow-time τ_n, and $s_{RC}[R(r, \phi, i), i, \tau_n]$ is the range compressed signal of the ith channel evaluated at the range $R(r, \phi, i, \tau_n)$. Notice that this range depends on the positions of the considered pixel (r, ϕ), on the position of the ith virtual element at the slow-time τ_n.

The focused image $I_{MIMO}(r, \phi, \tau_n)$ is a pass-band image in the wavenumber domain with the before-mentioned resolutions (Equations (1) and (2)).

The important detail is that by back-projecting the range compressed data on a *fixed* polar grid, each low-resolution image is already coregistered, the range migration has been compensated, and thus so has the phase curvature. We remark also that the back-projection is able to cope with any non-linear trajectory of the moving vehicle [18].

The next step in the processing chain that will make it possible to obtain the well-focused high-resolution SAR image is the autofocus. Its objective is to estimate the trajectory residue (the one not estimated by the vehicle's navigation unit). In the literature, there are several autofocusing techniques [14,19–21]. We remark, however, that each autofocus algorithm is based on a first detection of stable targets from low-resolution images. In this case, the low-resolution images are the N_τ MIMO images themselves.

This is also the reason why in this article, we do not deal with frequency domain focusing algorithms: we need to have low-resolution images in order to estimate the residual motion. Once we have them, we look for ways to combine them in the most efficient way.

After the residual motion estimation and compensation, the final SAR image can be formed. We identified three possible techniques: namely, the Fast Factorized Back-Projection (FFBP), 3D2D, and a derivation of the latter called Quick&Dirty (Q&D) [22]. In

the following sections, the three methods are explained, their limitations and pitfalls are detailed, and they are compared in terms of performances.

3. SAR Imaging: Proposed Methods

The SAR processing foresees the coherent summation of each low-resolution image to obtain an SAR image at fine resolution:

$$I_{SAR}(r,\phi) = \sum_{\tau_n=1}^{N_\tau} I_{MIMO}(r,\phi;\tau_n) \tag{5}$$

Before the summation can happen, each low-res image must be demodulated, interpolated on a finer grid to accommodate for the expansion of the bandwidth induced by the coherent summation, and re-modulated.

In the following sections, we describe the two algorithms in detail, providing also a mathematical interpretation. The assumptions behind each algorithm are detailed, and the limits of each are indicated.

3.1. FFBP Scheme

The FFBP algorithm [17] starts from the set of N_τ low-res images discussed in Section 2 and proceeds by interpolating sub-images in the angular domain to fine resolution before implementing the summation in Equation (5). The block diagram of the processing is depicted in Figure 3, and all the steps are here described:
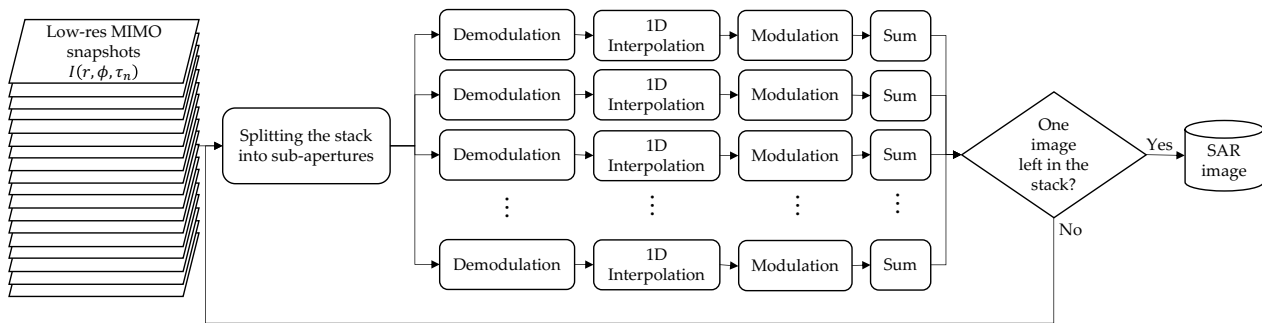

Figure 3. Block diagram of the FFBP algorithm.

1. The N_τ images are divided into sub-groups composed each by N_{sub} images. This leads to the generation of $L = N_\tau/N_{sub}$ sub-groups. Notice that each sub-group corresponds to a sub-aperture (i.e., to a portion of the total SAR aperture).
2. Each of the N_τ images is demodulated to bring them in base-band in the spatial frequency domain, avoiding in this way aliasing. This operation can be performed rigorously:

$$I_{MIMO}^{BB}(r,\phi;\tau_n) = I_{MIMO}(r,\phi;\tau_n) \times \exp\left\{-j\frac{4\pi}{\lambda}R(r,\phi,\tau_n)\right\} \tag{6}$$

    where $R(r,\phi,\tau_n)$ is the distance from the center of the virtual array to the pixel at coordinates (r,ϕ) at time τ_n.
3. Each image is now interpolated on a finer angular grid. The sampling of such a fine grid is computed by taking into account the current sub-aperture length (i.e., the PRI and the velocity of the vehicle). The interpolation is necessary to handle correctly the expansion of the bandwidth that will be generated in the following step.

$$I_{MIMO}^{BB}(r,\phi_{fine};\tau_n) = I_{MIMO}^{BB}(r,\phi \to \phi_{fine};\tau_n) \tag{7}$$

    where $\phi \to \phi_{fine}$ represents the 1D interpolation in the angular direction on a finer grid.

4. Every image is now modulated in order to bring them back into bandpass. The operation can be completed again rigorously as:

$$I_{MIMO}(r, \phi_{fine}; \tau_n) = I_{MIMO}^{BB}(r, \phi_{fine}; \tau_n) \times \exp \left\{ +j\frac{4\pi}{\lambda} R(r, \phi_{fine}, \tau_n) \right\} \quad (8)$$

5. All the images belonging to a sub-aperture are now coherently summed together, leading to a stack of L images with higher resolution with respect to the original low-resolution stack.

$$I_{SAR}^{MID}(r, \phi_{fine}, l) = \sum_{\tau_n \in \Omega_l} I_{MIMO}(r, \phi_{fine}; \tau_n) \quad (9)$$

where MID stands for mid-resolution, meaning that the whole synthetic aperture has not been exploited yet. l is the index of the sub-aperture ($l \in [1, L]$), and the summation for the lth sub-aperture is performed over all the images within that sub-aperture time Ω_l.

6. All the steps described up to now are repeated with the only difference that we do not start anymore by the N_τ MIMO images but from the L mid-resolution SAR images, and the distances used for modulation and demodulation are computed from the center of the sub-aperture to each pixel and no more from the center of the virtual array.

The critical step of this procedure is the interpolation in (7). We remark that while the car travels, the grid in Figure 2 is fixed in space and defined only once for the entire synthetic aperture. The consequence is that the grid is strictly polar just for the low-resolution image having the array center co-located with the origin of the grid. As an example, we can take Figure 2: in this case, just the last low-res image is back-projected on a grid that is truly polar. The previous three samples depicted in light red are back-projected on the same grid that is, however, not polar for them.

The result is an expansion of the bandwidth of the signal in the angular direction ϕ that in turn leads to a possible error during interpolation in (7). In the following, a theoretical treatment of this bandwidth expansion is proposed.

3.1.1. Limits and Drawbacks

Let us suppose to have a generic sensor placed in the origin of a Cartesian reference frame and a generic roto-translated polar grid as shown in Figure 4.
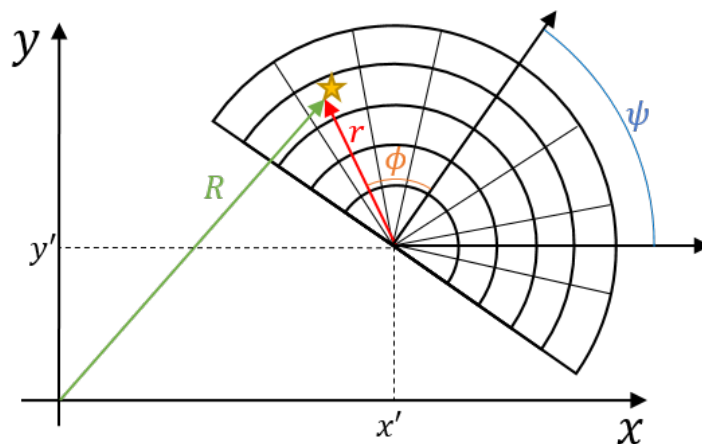

Figure 4. Roto-translated back-projection polar grid. The grid is centered in coordinates x' and y', and it is rotated with an angle ψ with respect to the Cartesian reference. The TX/RX antenna is located in the origin of the Cartesian reference. The target is represented by the yellow star.

By definition, the wavenumbers in the polar reference frame are defined as the partial derivatives of the distance in the range and angular direction. For a monocromatic wave, this leads to:

$$R = \sqrt{(x' + r\cos(\phi - \psi))^2 + (y' + r\sin(\phi - \psi))^2} \tag{10}$$

$$k_r = \frac{4\pi}{\lambda}\frac{\partial R}{\partial r} = \frac{4\pi}{\lambda}\frac{x'\cos(\phi - \psi) + y'\sin(\phi - \psi) + r}{R} \tag{11}$$

$$k_\phi = \frac{4\pi}{\lambda}\frac{\partial R}{\partial \phi} = \frac{4\pi}{\lambda}\frac{ry'\cos(\phi - \psi) - rx'\sin(\phi - \psi)}{R} \tag{12}$$

where x' and y' are the Cartesian coordinates of the polar grid origin, ψ is the rotation of the polar grid with respect to the Cartesian grid, and r and ϕ are the range and angular coordinates of the pixel in the polar grid.

Notice that, as expected, if the polar grid has the origin in the antenna phase center ($x' = y' = 0$, thus $R = r$), the wavenumber in the range direction is $4\pi/\lambda$, while in the angular direction, it is zero. These considerations are easily extended for a wideband signal and for a virtual array composed of more than one antenna.

It is straightforward to see that a roto-translation of the polar back-projection grid will correspond to a bandwidth expansion of the signal in the angular direction, since k_ϕ in Equation (12) is no more zero. The angular sampling of the polar BP grid, however, is not designed to handle this bandwidth expansion; thus, aliasing occurs.

We deem that the solution to this problem is to wisely choose the polar BP grid. In particular, it is better to chose a grid that is as close to be polar as possible for all the antenna phase centers. One way to do so is to fix the origin of the grid at the center of the aperture. The quality of the image will also depend on the interpolation kernel used. Some tests have been carried out, and the results will be shown in Section 4.

3.2. 3D2D Scheme

In this section, the 3D2D algorithm is detailed, and its limitations are explained. A block diagram of the algorithm is depicted in Figure 5.
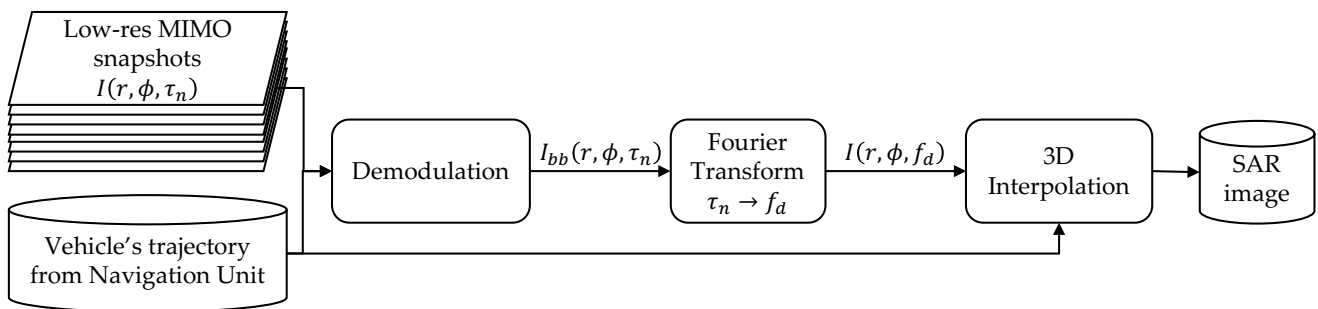

Figure 5. Block diagram of the 3D2D algorithm.

The algorithm starts again, as the FFBP, from the set of N_τ low-resolution images discussed in Section 2 and proceeds as follows:

1. Each low-res image is demodulated into baseband using a linear law of distances. We use the linear approximation due to the fact that later on, a Fourier transform will be used, and the FT consists of applying a linear phase to the data and integrating it.

$$I_{bb}(r, \phi, \tau_n) \approx I_{MIMO}(r, \phi, \tau_n) \times \exp\left\{-j\frac{4\pi}{\lambda}R(r, \phi, \tau_n)\right\} \tag{13}$$

where we use the $\approx$ and not the $=$ sign just to remember that the data are placed into baseband with an approximated law of distances given by:

$$R(r, \phi, \tau_n) = R_0(r, \phi) + v_r(\phi)(\tau_n - \tau_0) \tag{14}$$

In Equation (14), the parameter $R_0(r, \phi)$ represents the distance from the center of the aperture to the pixel in (r, ϕ), τ_0 is the slow-time of the center of the aperture, and $v_r(\phi)$ is the radial velocity of the car with respect to a fixed target in (r, ϕ). This velocity is calculated exploiting the nominal trajectory provided by the NU, and it is calculated at the center of the aperture.

Notice that this is just a linear approximation of the true hyperbolic range equation; the linearity assumption holds for short aperture and/or far-range targets. An analysis of the validity of this assumption will be provided to the reader in Section 3.2.2.

2. The Fourier transform (FT) in the slow-time dimension of the baseband data is taken:

$$I\left(r, \phi, f_d = -\frac{2}{\lambda} v_r\right) = \mathcal{F}\{I_{bb}(r, \phi, \tau_n)\} \tag{15}$$

where v_r is the radial velocity and $\mathcal{F}$ is the Fourier transform in the slow-time direction. This leads to a dataset in the range, angle, and Doppler frequency domain that can be easily converted in range, angle, radial velocity. The output is the 3D Range–Angle–Velocity (RAV) data cube. Notice that the Fourier transform simply multiplies the data by a linear phase term and integrates the result; thus, we can expect that in the $I(r, \phi, v_r)$ data cube, the final SAR image is present and must be simply extracted. We remark that the number of frequency points over which the FT is computed is an arbitrary parameter and can be chosen to provide a very fine sampling in the frequency (velocity) domain.

3. The extraction of the SAR image is carried out by interpolating the 3D cube over a 2D surface. The $(r_{fine}, \phi_{fine}, v_r^{NAV})$ coordinates of the 2D surface are calculated considering the high-resolution nature of the final SAR image and the nominal radial velocity provided by the vehicle's navigation unit:

$$I_{SAR}(r_{fine}, \phi_{fine}) = \mathcal{I}\{I(r, \phi, v_r); r_{fine}, \phi_{fine}, v_r^{NAV}\} \tag{16}$$

where $\mathcal{I}$ is the interpolation operation, r_{fine} and ϕ_{fine} are the range and angle coordinates of the final SAR image, and v_r^{NAV} is the radial velocity of each pixel in the fine resolution grid as calculated by navigational data.

Very interestingly, the 3D2D approach in Equation (16) has a precise geometrical meaning in that it describes the formation of an SAR image in terms of the intersection of the *curved* surface v_r^{NAV} with the 3D RAV data cube. Some Doppler layers of the RAV cube are represented in Figure 6. These layers are the common result of classical automotive radar processing. In particular, the maximum of the absolute value is usually taken in the Doppler domain to detect targets. The maximum is indeed taken in Figure 7 and plotted at the base of the image. The curved surface extracted from the RAV cube is also represented in Figure 7.
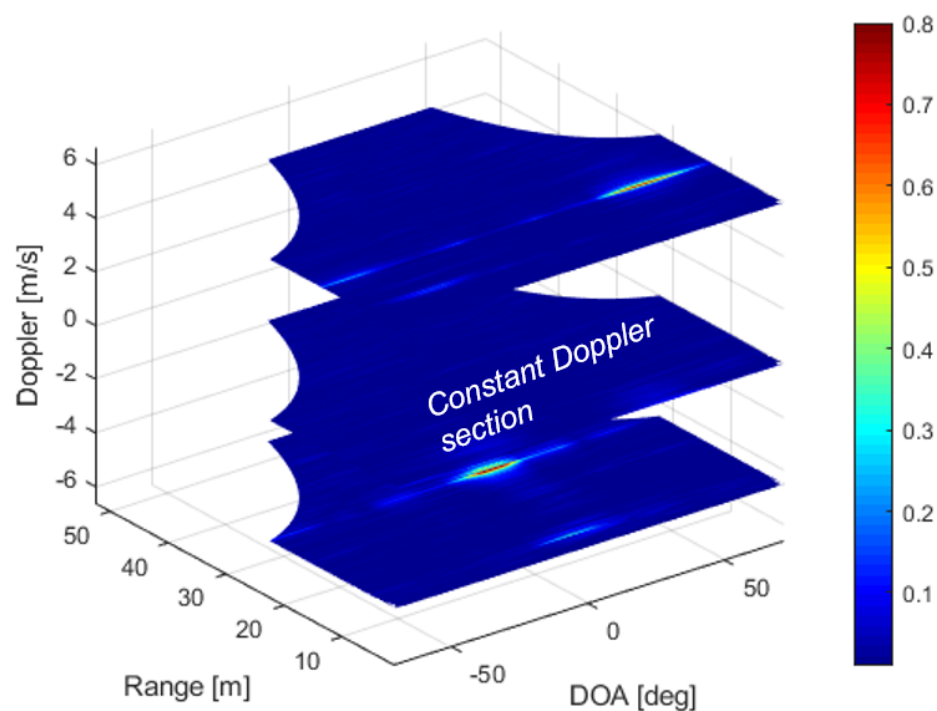
Figure 6. Range-Angle-Doppler data cube with three constant Doppler planes depicted.
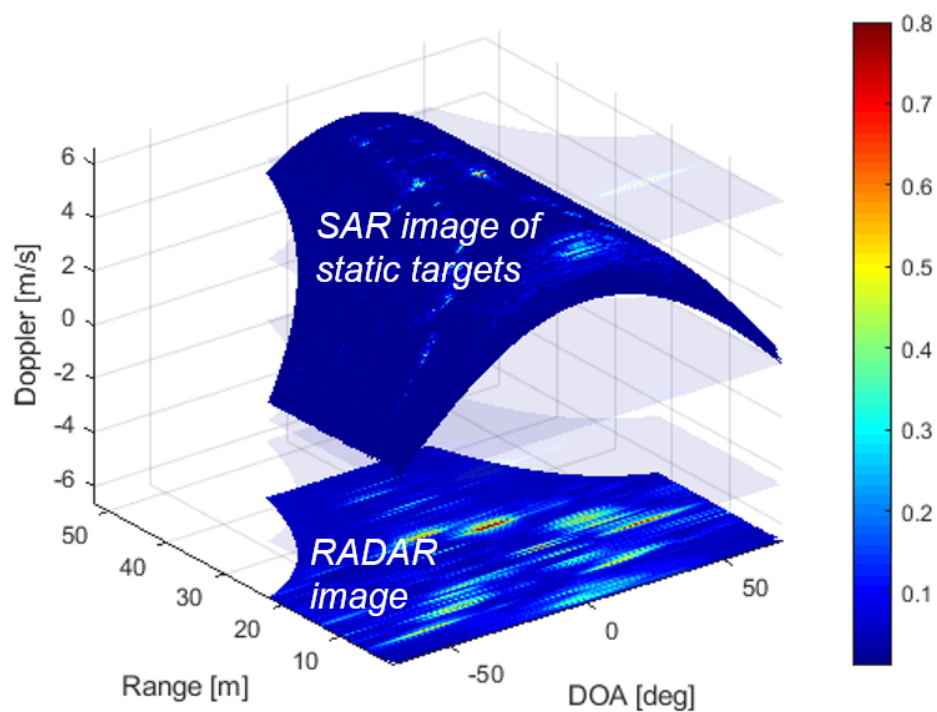
Figure 7. On the bottom of the plot, the radar image is computed by detecting the maximum Doppler for each range-angle bin. The SAR image can be interpreted as an intersection of the 3D RAV cube with a 2D surface.

3.2.1. Quick&Dirty Scheme (Q&D)

In this section, we present an alternative to the 3D2D approach that, however, exploits the same principle and follows the same geometrical interpretation of the 3D2D [22]. The block diagram is depicted in Figure 8.

The algorithm proceeds via Equation (16), exactly as the 3D2D approach, with the fundamental difference that the sub-images are generated without accounting for phase curvature and range migration. Such simplification results in the possibility to substitute the back-projection operator in Equation (4) with a simple FFT.
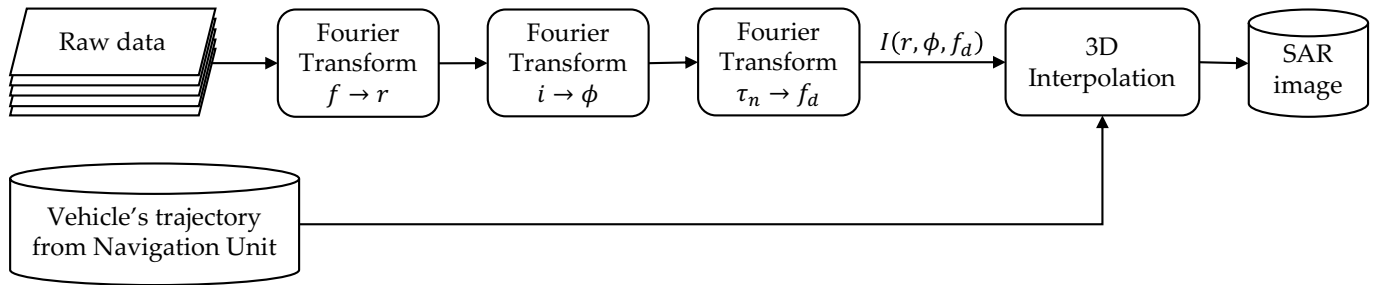

Figure 8. Block diagram of the Quick&Dirty (Q&D) algorithm.

The raw data are modeled as:

$$s(f, i, \tau_n) = \exp\left\{-j\frac{4\pi}{c}(f + f_0)R(i, \tau_n)\right\} \qquad (17)$$

where i is again the index of the virtual channel, f_0 is the carrier frequency, c is the speed of light and $R(j, \tau_n)$ is the distance from the target to the ith virtual array at slow-time τ_n. This last parameter can be approximated to the first order with respect to i and τ_n yielding to:

$$R(i, \tau_n) = R + id\sin\phi + v_r\tau_n \qquad (18)$$

where R, ϕ, and v_r are the distance, angular position, and radial velocity of the target with respect to the sensor at position $\tau_n = 0$, and d has already been defined as the distance between the virtual phase centers of the MIMO array.

Plugging Equation (18) into Equation (17) and applying the narrow bandwidth approximation, we obtain:

$$s(f, i, \tau_n) = \exp\left\{-j\frac{4\pi}{c}(fR + f_0 id\sin\phi + f_0 v_r\tau_n)\right\} \qquad (19)$$

It is easy to see that by transforming the data with respect to f, i and τ_n, we can resolve the target in range (r), angle (ϕ) and radial velocity (v_r, or equivalently in Doppler Frequency f_d).

In doing so, the generation of the RAV data cube is only obtained by FFT, as in conventional radar processing. Such an approach is clearly inaccurate and only feasible for short apertures. Yet, it possibly provides the fastest possible implementation of SAR imaging while also allowing for re-using RAV data produced in a conventional radar processing chain. In the next section, we provide the limits of applicability of this algorithm.

3.2.2. Limits and Drawbacks

In the FFBP, several 2D interpolations are carried out and, as shown in Section 3.1.1, an error in the interpolation may arise due to the bandwidth expansion induced by the roto-traslation of the polar back-projection grid.

For what concerns the 3D2D algorithm, the reasoning of Section 3.1.1 about the spatial bandwidth expansion can be applied here also. The interpolation, however, in this case happens in the 3D space (range, angle, and velocity).

The biggest limitation of this approach, however, is the one represented by the linear phase used for the demodulation of the low-resolution images. We recall that a linear phase is necessary, since the following step is a Fourier transform that works by applying linear phases with different slopes to the data and sums the result. The issue is that the linear phase is just an approximation of the real phase history of a target in the scene. An example is useful to clarify this topic. Given a platform traveling in the x direction at a speed v_x, we have that the range equation can be written as:

$$R(\tau_n) = \sqrt{R_0^2 + (v_x\tau_n)^2} \approx \underbrace{R}_{\text{constant}} + \underbrace{\cos(\phi)v_x\Delta\tau}_{\text{linear}} + \underbrace{\frac{v_x^2\sin^2(\phi)}{R}\Delta\tau^2}_{\text{parabolic}} \tag{20}$$

where the approximation is the second-order Taylor expansion of the range equation around a slow time $\tau_n = \tau_0$, R is the distance from the center of the considered aperture (the position of the antenna phase center at slow-time τ_0), ϕ is the squint angle as in Figure 1a and $\Delta\tau = \tau_n - \tau_0$.

Notice that the parabolic component is equivalent to a chirp function that leads to an expansion of the bandwidth after the Fourier transform in (15).

The parabolic component of the range can be neglected when it is bounded by half the wavelength at its maximum—that is, for $\Delta\tau = T_a/2$. This condition leads to a constraint for the maximum aperture length:

$$\frac{v_x^2\sin^2(\phi)}{R}\left(\frac{T_a}{2}\right)^2 = \frac{\sin^2(\phi)A_s^2}{4R} \leq \frac{\lambda}{2} \quad \longrightarrow \quad A_s \leq \sqrt{\frac{2\lambda R}{\sin^2(\phi)}} \tag{21}$$

For a target at $\phi = 45$ deg and a close range of $R = 15m$, the maximum aperture length is $A_s = 48$ cm. If the aperture length is bigger, the linear approximation starts to lose validity; thus, the images are not correctly placed in the baseband. This leads in turn to a defocusing of the image. In the following section, we simulate different trajectories in order to assess the quality of the Impulse Response Function (IRF) under different conditions.

To provide a quantitative evaluation of the limits of the Q&D algorithm, we derive here an upper bound on the coherent integration time that is allowed without explicitly accounting for range migration and phase curvature. As for range migration, the requirement to set is that range variations do not exceed range resolution:

$$v_x cos(\phi)T_a < \frac{c}{2B} \tag{22}$$

Noticing that $A_s = v_xT_a$, we obtain that for a target at $\phi = 45$ deg and a bandwidth of $B = 1$ GHz, the maximum aperture length is roughly 20 cm.

The limit on the phase curvature is the same already described for the 3D2D (see Equation (21)).

4. Point Target Analysis—Results With Simulated Data

In this section, we test the performances of the algorithms described before in terms of Impulse Response Function (IRF) quality and computation time. The simulations will be carried out for several aperture lengths for better understanding the limitations given by each approach. The simulated scenario is represented by a vehicle moving on a straight trajectory. The radar has eight virtual channels transmitting a chirp with 1 GHz of bandwidth with a Pulse Repetition Frequency of 7 KHz. The overall configuration (summarized in Table 1) is trying to resemble a real system such as the one in [13,14].

From now on, we will focus on the FFBP and 3D2D and not on the Q&D. This latter, in fact, can be seen as an inaccurate 3D2D where we accept to lose quality of the image in favor of a simpler and faster processing scheme.

Table 1. Parameters of the simulated radar and vehicle.

Parameter	Value
Bandwidth (B)	1 GHz
Number of virtual channels (N_{ch})	8
PRF	7 KHz
Number of pulses forming an aperture	256
Simulated vehicle velocities (v)	5, 30, 40, 50 m/s
Mode	Forward looking

In the simulation, a target has been placed at position $x_0 = y_0 = 10$m. A direct TDBP was performed on a very fine but small polar grid centered around the nominal position of the target. The sampling steps in range (r) and angle (ϕ) of this grid correspond to one-tenth of the respective resolutions, namely:

$$\Delta r = \frac{1}{10} \frac{c}{2B} \tag{23}$$

and

$$\Delta\phi = \frac{1}{10} \frac{\lambda}{2A_s} \tag{24}$$

where A_s is the length of the synthetic aperture. Notice that we took the *maximum* angular resolution.

This kind of processing is not efficient for practical applications, since the required computational burden is too high when the imaged scene (i.e., the size of the BP grid) becomes large. Nevertheless, this approach is used to generate the images that will be used as a benchmark for the two focusing algorithms.

4.1. Choosing the Interpolation Kernel

We begin the comparison by considering a rather small aperture of 18 cm. The objective of this section is to compare several interpolation kernels to assess which one provides the best imaging quality. The simulated vehicle is traveling at 5 m/s (18 km/h), transmitting a burst of 256 pulses with a PRF of 7 KHz. Such an aperture leads to a maximum angular resolution in the direction orthogonal to motion of 0.6 deg.

The interpolation in Equation (4) has been tested using several interpolation kernels, and it is depicted in Figure 9. In this image, we focused the response from a point target immediately in the fine resolution polar grid. From now on, a perfectly focused IRF will have a unitary peak value. As expected, the higher the order of the interpolator, the better the target is resolved both in terms of peak value and side lobes. The best interpolator is using the cardinal sine kernel; however, this is by far the most computationally demanding filter to be used.

For the comparison of the two algorithms, we start by a common set of low-resolution MIMO images. Each snapshot is formed by TDBP by using a very coarse polar grid with the origin that coincides with the center of the synthetic aperture. The sampling spacing in the range (r) and angle (ϕ) of this grid corresponds to half of the respective resolutions, namely:

$$\Delta r = \frac{1}{2} \frac{c}{2B} \tag{25}$$

and

$$\Delta\phi = \frac{1}{2} \frac{\lambda}{2N_{ch}d} \tag{26}$$

This choice has been made to keep the time required for the generation of the snapshots low.
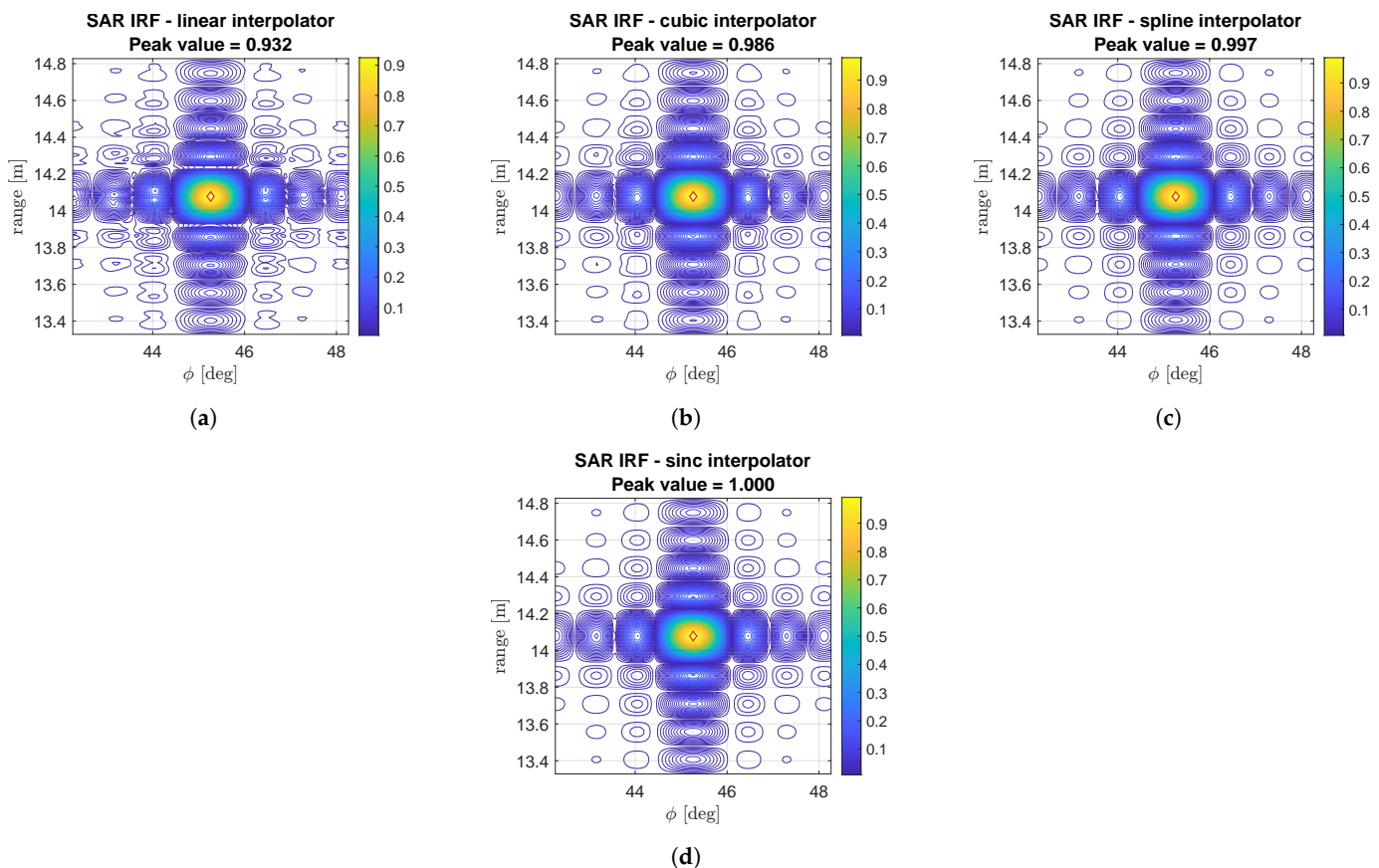
(**a**)   (**b**)   (**c**)

(**d**)

Figure 9. $A_s = 18$ cm. SAR Impulse Response Function (IRF) generated by direct TDBP on a very fine polar grid centered in the nominal position of the target. The higher the order of the interpolator, the better the image in terms of peak value and side lobes.

In Figure 10, the FFBP has been tested using different interpolation kernels. In addition, in this case, the quality depends on the interpolation kernel used. The higher the degree of the interpolator, the better the image. The spline interpolator reaches practically the same accuracy of the direct TDBP.
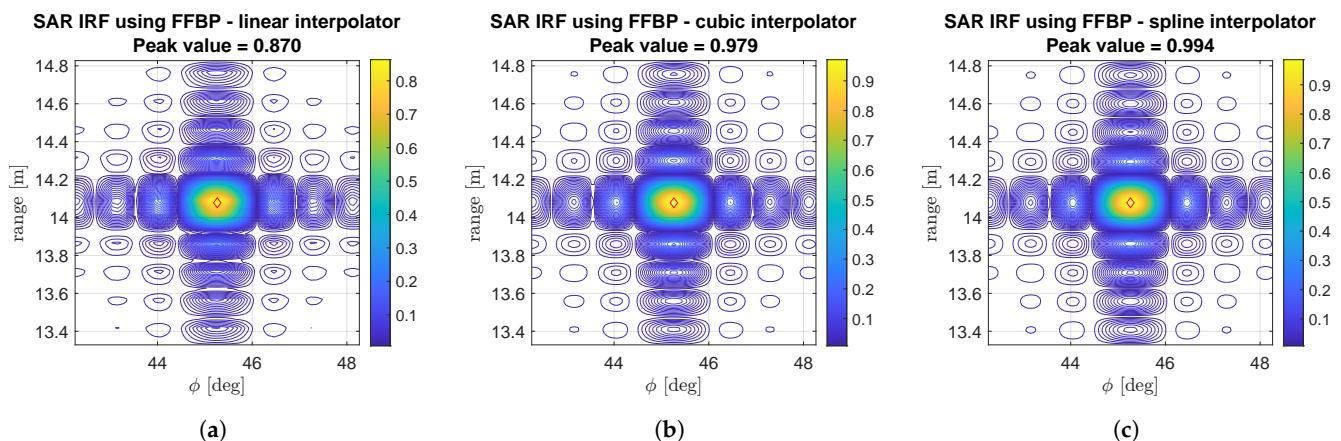

(**a**)   (**b**)   (**c**)

Figure 10. $A_s = 18$ cm. SAR ImpulseResponse Function (IRF) generated by FFBP on a very fine polar grid centered in the nominal position of the target. The higher the order of the interpolator, the better the image in terms of peak value and sidelobes.

The same considerations on the interpolation kernel are valid for the 3D2D algorithm for which the IRF is depicted in Figure 11. In this case, since the aperture is rather small, the linear approximation of Equation (14) is valid; thus, the images are correctly placed in

the baseband, and a correct imaging can take place. Using a cubic interpolator, the peak value of the IRF is just 0.11 dB lower than the optimal one.
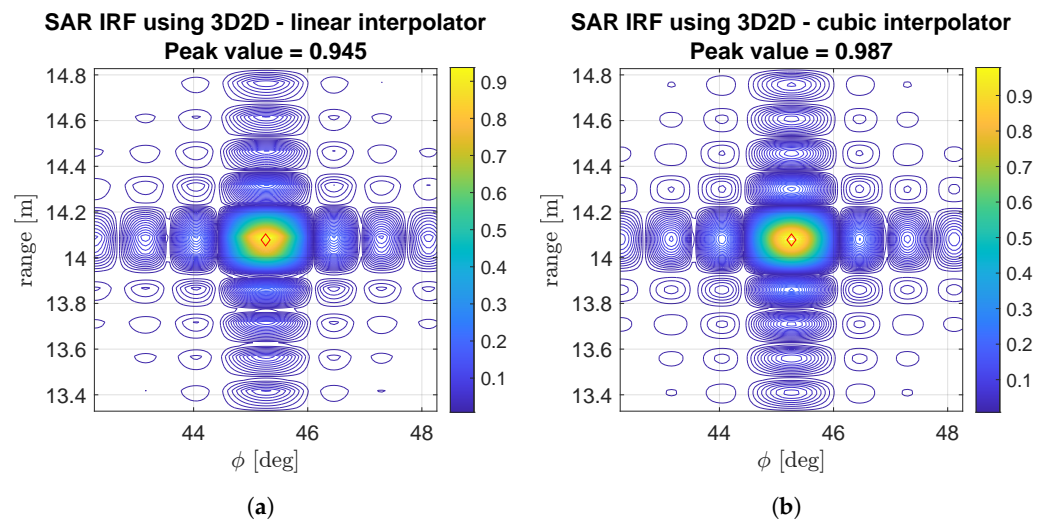

(**a**)   (**b**)

Figure 11. $A_s = 18$ cm. SAR Impulse Response Function (IRF) generated by 3D2D on a very fine polar grid centered in the nominal position of the target. The higher the order of the interpolator, the better the image in terms of peak value and sidelobes.

Regarding the image quality, we deem that for short apertures, the two algorithms are comparable. If the processing units onboard the vehicle allow a fast implementation, it is better to use high-order interpolation kernels such as the spline and cubic interpolator.

4.2. Imaging with Longer Apertures

In this section, we compare again the two algorithms for different aperture sizes. The target is once again placed in $x_0 = y_0 = 10$m. In order to simulate different aperture sizes, we change the velocity of the vehicle while maintaining the PRF of the radar system as fixed. The tested velocities are:

- $v = 30$ m/s (108 km/h) leading to an aperture of 1.06 cm;
- $v = 40$ m/s (144 km/h) leading to an aperture of 1.45 m;
- $v = 50$ m/s (180 km/h) leading to an aperture of 1.82 m.

In Figure 12, the three experiments are represented on each row, while the columns represent the different algorithms under test. For all the simulations, we used a cubic interpolator, since it is the one showing better performances.

From Figure 12a,d,g, we see how the direct BP preserves the image quality even with longer apertures. This is an expected result, since the Time Domain Back-Projection is an exact algorithm that properly takes into account the range migration and non-linear nature of the phase.

Almost the same can be said for the FFBP (Figure 12b,e,h). This algorithm always uses the true distances for demodulating and modulating the data; therefore, no image degradation happens for longer apertures. The 3D2D, on the other hand, is not able to properly handle the non-linearity of the phase with long apertures. The effect is clear from Figure 12i where IRF is totally smeared.

In Table 2, the normalized peak values of the IRF for each experiment are summarized.
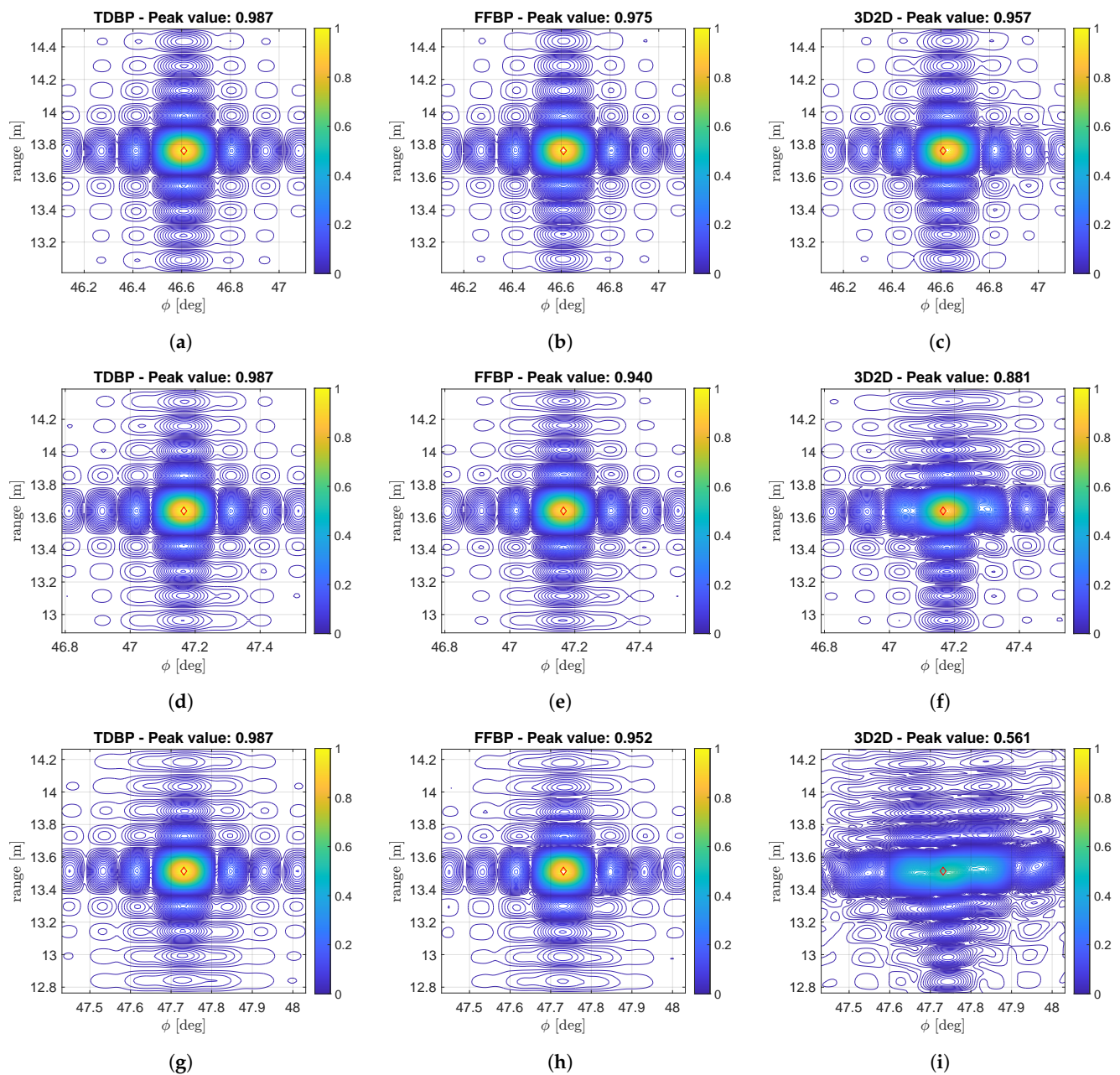
Figure 12. IRF with different aperture size. (First row) $A_s = 1.06$ m. (Second row) $A_s = 1.45$ m. (Third row) $A_s = 1.82$m. All the IRF are focused on using a cubic interpolator.

Table 2. Normalized peak values of the IRF for different velocities (aperture length) and for different focusing algorithms.

	$v = 30$ **m/s**	$v = 40$ **m/s**	$v = 50$ **m/s**
Direct TDBP	0.987	0.987	0.987
FFBP	0.975	0.940	0.952
3D2D	0.957	0.881	0.561

The longer the aperture, the more the linear law used to baseband the low-resolution images is invalid. The approximated law of distances is more and more an approximation while we move at the extreme of the aperture. In Figure 13, some demodulated spectra of low-resolution images in the stack are depicted. In Figure 13a,d, the image at the center of the aperture is demodulated using the correct (hyperbolic) law and the approximated (linear) law of distances. It is clear that the two are exactly the same as expected. While we

move toward the extremes of the aperture, the two spectra became more and more different, with a significant distortion happening for the ones demodulated using the approximated distance (see Figure 13e,f).
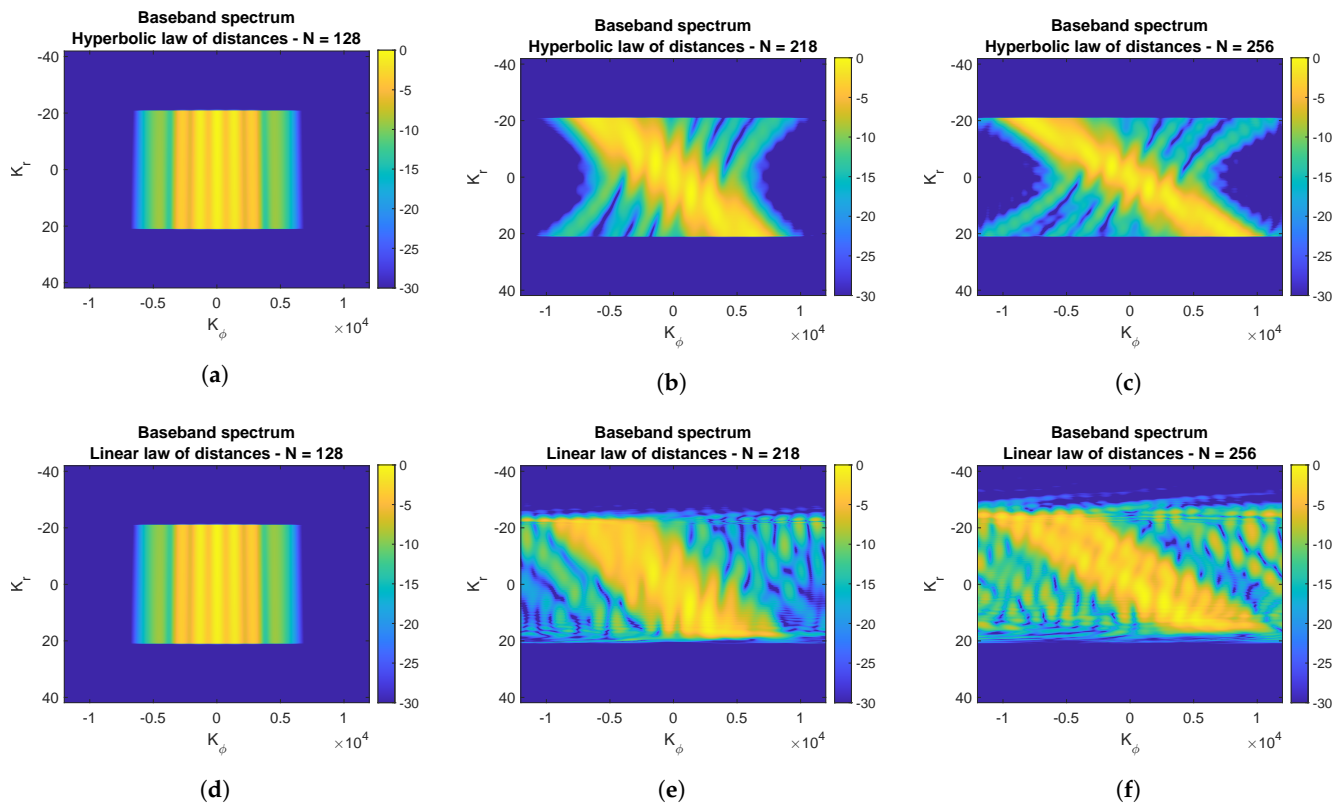

Figure 13. Baseband spectra of some low-resolution MIMO images. The row on top shows the spectra demodulated with the true (hyperbolic) law of distances. The row on the bottom shows the spectra demodulated with the approximated (linear) low of distances. In the left column, the spectra of the image in the middle of the aperture are represented. In this case, the linear approximation matches perfectly the true distance. Moving away from the center of the aperture (second and third columns), the spectra start to be more and more aliased, leading to a degradation of the final SAR image.

Figure 12 provides a first quality metric for the IRF: the peak value. In Table 3, the Integrated Side Lobes Ratio (ISLR) is also depicted. The ISLR [15] is defined as:

$$ISLR = 10 \log_{10} \left(\frac{P_{\text{total}} - P_{\text{main}}}{P_{\text{main}}} \right) \tag{27}$$

where P_{total} is the total power of the IRF, while P_{main} is the power inside the main lobe.

The figure of merit confirms what is already clear: the 3D2D algorithm worsens for long apertures, reaching unacceptable values of ISLR for speed between 40 and 50 m/s.

Table 3. ISLR for different velocities (aperture length) and for different focusing algorithms.

	$v = 5$ m/s	$v = 30$ m/s	$v = 40$ m/s	$v = 50$ m/s
Direct BP	−12.1 dB	−12.17 dB	−12.21 dB	−12.21 dB
FFBP	−12.2 dB	−12.24 dB	−12.37 dB	−12.48 dB
3D2D	−12.16 dB	−12.15 dB	−10.46 dB	−6.63 dB

We remark that all the proposed simulations use a single point target at almost 14 m of range. For the same angle ϕ, but for longer ranges r, the assumption on the phase linearity may hold even for long apertures, as shown in Equation (21); thus, the image will be focused properly.

To conclude this section, we make some practical considerations for applying such algorithms in the automotive industry. Both algorithms work as expected for short aperture, leading to a well-focused and well-localized IRF. This is no longer the case for more extended aperture, since the 3D2D quickly loses the linear phase assumption, leading to bad performances. We deem, however, that such extreme aperture lengths are very often unnecessary: a synthetic aperture of just 0.5 cm leads to a resolution in the order of 0.3 deg around 45 deg, which is comparable with commercial LiDARs used in the automotive industry.

If extreme resolutions are necessary, a possible solution to preserve the image quality using the 3D2D approach is to divide the aperture into small sub-apertures where the phase can be considered linear; then, we combine these sub-apertures to obtain the full-resolution image.

5. Computational Costs: Rough Number of Operations and Processing Time

We now derive the computational burden of the two methods in terms of Rough Number of Operations (RNO). This figure of merit is tightly related to the computational time requested to form a full-resolution SAR image.

The objective of this section is to show that both the algorithms are suitable for real-time urban mapping.

5.1. Range Compression

In FMCW radars, the range compression is performed by a simple fast Fourier transform (FFT) of the demodulated and deramped raw data [16]. The number of operations required by the FFT when the number of frequency points computed (N_f) is equal to a power of two is equal to $N_f \log_2 N_f$. The FFT must obviously be computed for each channel (N_{ch}) and for each slow-time (N_τ), leading to a total number of operations equal to:

$$N_{ch} \times N_\tau \times N_f \log_2 N_f \tag{28}$$

In the following, we will give some indicative numbers providing a quantitative indication of the number of operations required by this step of the workflow.

5.2. Low-Resolution MIMO Image Formation

The second step is the TDBP of the range compressed data to form the stack of low-resolution snapshots of the scene. For each pixel in the BP grid, we need to:

1. Calculate the bistatic distances from each real antenna phase center to the considered pixel. This operation is composed of six squares and two square roots (Pythagorean theorem) for a total of eight operations per channel.
2. Interpolate the range compressed data using a linear or nearest neighbor interpolator accounting for two operations per channel.
3. Modulate the signal, which is just a single complex multiplication per pixel.
4. Accumulate the back-projected signal for each channel with a single complex summation.

We remark that these operations must be computed for each pixel in the BP grid and for each slow-time; thus, the final number of operations is equal to:

$$N_r \times N_a \times N_\tau \times 12 N_{ch} \tag{29}$$

where N_r and N_a are the sizes of the polar BP grid in the range and angular direction, respectively, and the other symbols were already defined in the previous section.

5.3. FFBP Rough Number of Operations

As already discussed, the FFBP is divided into stages. In the first stage, we have N_τ slow-time snapshots, each one composed by N_a samples in angle and N_r samples in range. Each sub-aperture is composed by N_{sub} samples, leading to a set of $L = N_\tau / N_{sub}$ sub-apertures.

The rough number of operations to be performed at the first stage of the FFBP can be expressed as:

$$\frac{N_\tau}{N_{sub}} \times N_{sub} \times (N_{sub} N_a) \times N_r \times K \tag{30}$$

where

- N_τ / N_{sub} is the number of sub-apertures;
- N_{sub} is the number of samples for each sub-aperture;
- $N_{sub} N_a$ expresses the number of pixels in the output grid from this stage which is proportional to the number of samples in the input grid (N_a) and the number of samples for each sub-aperture (N_{sub});
- N_r is the number of pixels in range (it is constant since the interpolation is mono dimensional in angle);
- K is the number of operations required to form each of the $(N_{sub} N_a) \times N_r$ pixels in the output grid.

In the second stage, the number of samples for each sub-aperture remains the same, while the number of sub-apertures decreases by a factor of N_{sub}. What instead increases is the number of angular samples of the output grid, which again increases by a factor of N_{sub}. The resulting expression for the number of operations is then:

$$\text{OP} = \frac{N_\tau}{N_{sub}^2} \times N_{sub} \times (N_{sub}^2 N_a) \times N_r \times K \tag{31}$$

which is the same value of the first stage. It is now clear that the number of operations is constant for each stage of the FFBP, and it is equal to

$$N_\tau \times (N_{sub} N_a) \times N_r \times K \tag{32}$$

The number of stages depends on the size of the initial set of snapshots (N_τ) and on the size of each sub-aperture (N_{sub}); thus, the total number of operations needed to form the final SAR image is:

$$N_\tau \log_{N_{sub}} N_\tau \times (N_{sub} N_a) \times N_r \times K \tag{33}$$

In Section 5.5, we provide some realistic figures for each parameter of Equation (33). For what concerns the value of K, instead, we can retrace the FFBP workflow and make a rough estimate of the number of operations requested per pixel:

1. We first calculate the distance from the center of the virtual array to the pixel itself. This passage accounts for three squares and one square root (Pythagorean theorem).
2. Demodulate the low-resolution images using the distances calculated in the previous step. This accounts for a single complex multiplication.
3. Interpolate using a mono-dimensional interpolation in the angular direction that accounts for two or three operations, depending on the interpolation kernel.
4. Compute again the distances, this time on the finer output BP grid. In addition, this time, we have a total of four operations.
5. Modulate the interpolated images using the distances previously computed using a single complex multiplication.

By summing all the figures provided, we can state that the rough number of complex operations requested for each pixel is $K = 12$.

5.4. 3D2D Rough Number of Operations

The 3D2D algorithm starts again from the set of low-resolution MIMO images; thus, the number of operations required by the range compression and the initial TDBP are in common with the FFBP. The next step is a demodulation of each of the N_τ low-resolution images. For each pixel, a single multiplication is performed to compute the distance Equation (14) and a multiplication is performed to demodulate the data. This consideration leads to a number of operations equal to:

$$N_\tau \times N_a \times N_r \times 2 \tag{34}$$

The next step is a slow-time FFT to be computed over each pixel, leading to:

$$N_a \times N_r \times N_v \log_2 N_v \tag{35}$$

where N_v is the number of frequency (or velocity) points computed by the FFT, a typical value is $N_v = 8N_\tau$.

The final step is the 3D interpolation. For each pixel of the output fine resolution polar grid, we have to compute the radial velocity (see Equation (16)). This step requires a number of operations equal to the size of the output grid $N_r \times N_{ah}$ where $N_a h$ is the size in the angular direction of the fine resolution polar grid.

Now, the interpolation takes place with a number of operations in the order of $27 \times N_r \times N_{ah}$. The number 27 is derived from the fact that the current pixel is derived by combining the 27 closest neighbors, i.e., the $3 \times 3 \times 3$ cube centered on the pixel.

These considerations lead to a total number of operations equal to:

$$\underbrace{N_\tau \times N_a \times N_r \times 2}_{\text{demodulation}} + \underbrace{N_a \times N_r \times N_v \log_2 N_v}_{\text{FFT}} + \underbrace{28 \times N_r \times N_{ah}}_{\text{interpolation}} \tag{36}$$

5.5. Comparison

In Table 4, we provide some realistic values for the parameters described in the previous sections.

Table 4. List of the parameters involved in the calculation of the total number of operations requested to form an SAR image.

Parameter	Description	Typical Value
N_t	Number of fast time samples	512
N_f	Number of frequency samples after range compression	1024
N_τ	Number of slow-times samples for each aperture	256
N_{ch}	Number of MIMO channels	8
N_r	Number range samples in the polar BP grid	400
N_a	Number angular samples in the low-res polar BP grid	17
N_{ah}	Number angular samples in the fine resolution polar BP grid	2048
N_{sub}	Sub-aperture size	2
N_v	Number of velocity points for 3D2D	2048

With these values, we can provide some quantitative results on the number of operations required by the two algorithms. We remark once again that it is not our intention to provide a accurate figure but rather just a rough order of magnitude.

Table 5 shows the number of operations (in millions, MM) for each one of the two algorithms.

Table 5. Rough number of operations required to form an SAR image using the two described algorithms.

Processing Block	Range Compression	TDBP	SAR Focusing	Total
FFBP	21 MM	148 MM	298 MM	467 MM
3D2D	21 MM	148 MM	159 MM	329 MM

Both the algorithms stay below half a billion of operations per image. For comparison, a 9-year-old iPhone®5S has a raw processing power of 76 GigaFlops, meaning it could focus an SAR image in 6 ms. We deem that with dedicated hardware such as FPGA/GPU, the urban scenario can be easily imaged in real time.

5.6. Processing Time on a Reference Machine

In this case, instead of focusing a small patch around the position of the target, we focus the entire visible scene as seen by the radar. In other words, the final polar grid will span an angle ϕ between -90 deg and $+90$ deg, and the range will go from the very near range ($r = 0$ m) to the far range ($r = 40$ m).

We used as a reference machine a Dell ®Precision 5820 with a 6 Core Intel Xeon W-2133. A summary of the computational performances is detailed in Table 6. In this table, we compare the two algorithms with different aperture sizes N_τ. In the parentheses, there is the processing time requested to form the low-resolution MIMO images, while it is outside the time requested by the algorithm itself.

With $N_\tau = 256$, the required time is basically the same, while for $N_\tau = 512$, the performances of the FFBP degrade, unlike the 3D2D that maintains almost the same processing time.

We remark also that the 3D2D can take advantage by fast GPU-based implementation of the FFT and 3D interpolation. Running the algorithm on an Nvidia Quadro P4000, we can cut the processing time by almost a factor of five.

This table includes all the best achievable performances with every algorithm (fastest interpolator and fastest sub-aperture size).

Table 6. Processing times for different algorithms. All the numbers are in seconds.

	$M = 256$	$M = 512$
Direct BP	72.93	148.82
FFBP	0.42 (+1.07)	0.68 (+2.02)
3D2D	0.43 (+1.07)	0.48 (+2.02)
3D2D (GPU)	0.11 (+1.07)	0.19 (+2.02)

These processing times must be taken as reference time and will not reflect the real performances of a deploy-ready implementation.

6. Experimental Results With Real Data

In this section, we show the results from a few experimental campaigns carried out using a 77 GHz MIMO radar mounted in forward-looking mode on an Alfa-Romeo®Giulia Veloce.

The data are acquired by operating in burst mode with a duty cycle of 50% and transmitting chirp pulses with a bandwidth of 1 GHz. The PRF is about 7 kHz. The car is also fully equipped with navigation sensors to provide a rough estimate of the trajectory. We remark once again that the trajectory provided by the navigation unit is not sufficiently accurate and must be complemented by an autofocusing routine. Only in this way is it possible to achieve very high focusing and localization accuracy. To show its focusing performances in a real scenario, all the results provided in this section have been focused using the 3D2D processing scheme.

In Figure 14, an example of processed aperture is depicted. On the top of the figure, an optical image is shown. At the bottom (central), an SAR image of the field of view is represented, while two zoomed details are shown on the left and the right. In the red rectangle, the parked cars to the vehicle's left are highlighted, while on the right, the same green car visible in the optical image is also recognizable in the SAR image. The SAR images' fine resolution allows recognition of even the smallest details, such as the green car's alloy wheels.

Figure 14. (**top**) Optical image of the field of view. (**bottom central**) SAR image of the entire scenario. (**bottom right**) A detail of the full SAR image: the green car in the optical image is parking. (**bottom left**) Another detail: some parked cars to the left of the vehicle.

Another example is the one depicted in Figure 15. This experiment was carried out to test the capability of SAR images to behave as expected even in complex situations such as when a pedestrian is present on the roadway very close to parked cars. A target with a very high Radar Cross-Section (RCS), such as a parked vehicle, can hide a target with a low RCS (such as a pedestrian). This is not the case in SAR images, where the excellent resolution of the system guarantees the detection of both targets. In Figure 15, a focused SAR image is shown on the left with some zoomed details on the bottom right and an optical image for reference. A row of parked cars is highlighted in the yellow box, while a free parking spot is depicted in the far field (green box). Notice how the resolution is degraded at boresight (i.e., in front of the car). The most important detail, however, is in the red box. A pedestrian is visible in the optical image, and it is easily detected also in the SAR image.
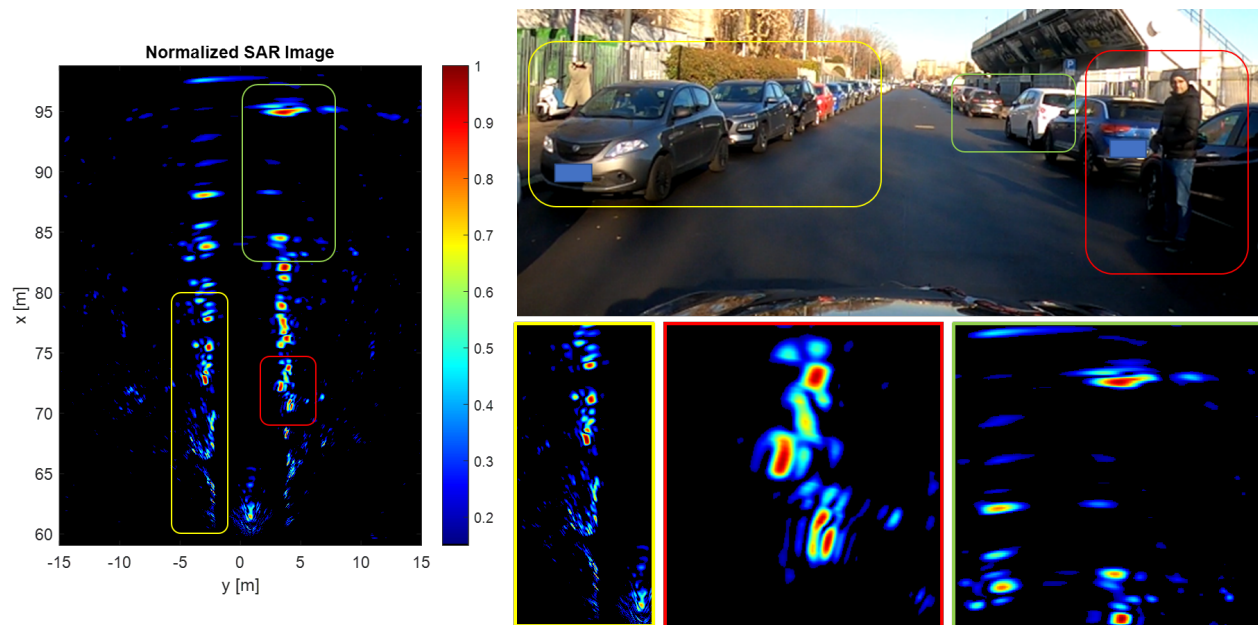
Figure 15. (**left**) SAR image of the field of view: (**top right**) Optical image. (**bottom right**) Three details of the SAR image highlighted with the some color in the optical image and in the full-field SAR image.

7. Conclusions

This paper aims to find the best algorithm to coherently combine low-resolution MIMO images to obtain a fine resolution SAR image. We deem that the generation of low-resolution images is a mandatory starting point for every vehicle-based SAR imaging algorithm. The reason is that such images allow for a simple and intuitive residual motion estimation, which in turn improves image quality and target localization accuracy.

We started with a theoretical review of the well-known Fast Factorized Back-Projection (FFBP), highlighting its strengths and weaknesses. We introduced the 3D2D scheme, detailing the algorithm itself and finding its theoretical limits regarding the aperture length. A possible alternative to the 3D2D called Quick&Dirty is also briefly described. It neglects range migration and phase curvature in favor of simpler processing.

We then proceed by making a comparison between the proposed schemes. We have compared several interpolation kernels and found that it is always better to use high-order interpolators such as cubic or spline, leading to higher-quality images with minimum additional computational cost.

We then compared the algorithms using longer apertures. In this scenario, the FFBP performs better than the 3D2D scheme. The reason is that while the former always uses the true law of distances for modulation/demodulation of the sub-images, the latter uses just a linear approximation. When apertures become long, the linearity hypothesis does not hold, leading to a deterioration of the image.

For the automotive industry, computational complexity is also a key factor. We provided a rough number of operations required by both processing schemes. The result is that both algorithms could run in real time if adequately implemented to work on dedicated hardware. The 3D2D/Q&D can also take particular advantage of GPU/FPGA implementation of the FFT.

While both algorithms are suited for a future real-world deployment, we deem that the 3D2D scheme is possibly simpler to implement, provides high-quality images, is generally faster, and can be easily transformed in an even faster scheme such as the Q&D. We proved that this processing scheme is suitable for an accurate imaging of the environment by showing results acquired in an open road campaign using a forward-looking 77 GHz MIMO radar.

References

1. Saponara, S.; Greco, M.S.; Gini, F. Radar-on-Chip/in-Package in Autonomous Driving Vehicles and Intelligent Transport Systems: Opportunities and Challenges. *IEEE Signal Process. Mag.* **2019**, *36*, 71–84. [CrossRef]
2. Ye, L.; Gu, G.; He, W.; Dai, H.; Lin, J.; Chen, Q. Adaptive Target Profile Acquiring Method for Photon Counting 3-D Imaging Lidar. *IEEE Photonics J.* **2016**, *8*, 1–10. [CrossRef]
3. Wang, P.; Wang, L.; Leung, H.; Zhang, G. Super-Resolution Mapping Based on Spatial–Spectral Correlation for Spectral Imagery. *IEEE Trans. Geosci. Remote Sens.* **2021**, *59*, 2256–2268. [CrossRef]
4. Li, J.; Stoica, P.; Zheng, X. Signal Synthesis and Receiver Design for MIMO Radar Imaging. *IEEE Trans. Signal Process.* **2008**, *56*, 3959–3968. [CrossRef]
5. Tebaldini, S.; Manzoni, M.; Tagliaferri, D.; Rizzi, M.; Monti-Guarnieri, A.V.; Prati, C.M.; Spagnolini, U.; Nicoli, M.; Russo, I.; Mazzucco, C. Sensing the Urban Environment by Automotive SAR Imaging: Potentials and Challenges. *Remote Sens.* **2022**, *14*, 3602. [CrossRef]
6. Stanko, S.; Palm, S.; Sommer, R.; Klöppel, F.; Caris, M.; Pohl, N. Millimeter resolution SAR imaging of infrastructure in the lower THz region using MIRANDA-300. In Proceedings of the 2016 46th European Microwave Conference (EuMC), London, UK, 4–6 October 2016; pp. 1505–1508. [CrossRef]
7. Gishkori, S.; Daniel, L.; Gashinova, M.; Mulgrew, B. Imaging for a Forward Scanning Automotive Synthetic Aperture Radar. *IEEE Trans. Aerosp. Electron. Syst.* **2019**, *55*, 1420–1434. [CrossRef]
8. Gishkori, S.; Daniel, L.; Gashinova, M.; Mulgrew, B. Imaging Moving Targets for a Forward Scanning SAR without Radar Motion Compensation. *Signal Process.* **2021**, *185*, 108110. [CrossRef]
9. Farhadi, M.; Feger, R.; Fink, J.; Wagner, T.; Stelzer, A. Automotive Synthetic Aperture Radar Imaging using TDM-MIMO. In Proceedings of the 2021 IEEE Radar Conference (RadarConf21), Atlanta, GA, USA, 7–14 May 2021; pp. 1–6. [CrossRef]
10. Laribi, A.; Hahn, M.; Dickmann, J.; Waldschmidt, C. Performance Investigation of Automotive SAR Imaging. In Proceedings of the 2018 IEEE MTT-S International Conference on Microwaves for Intelligent Mobility (ICMIM), Munich, Germany, 15–17 April 2018; pp. 1–4. [CrossRef]
11. Feger, R.; Haderer, A.; Stelzer, A. Experimental verification of a 77-GHz synthetic aperture radar system for automotive applications. In Proceedings of the 2017 IEEE MTT-S International Conference on Microwaves for Intelligent Mobility (ICMIM), Nagoya, Japan, 19–21 March 2017; pp. 111–114. [CrossRef]
12. Gao, X.; Roy, S.; Xing, G. MIMO-SAR: A Hierarchical High-Resolution Imaging Algorithm for mmWave FMCW Radar in Autonomous Driving. *IEEE Trans. Veh. Technol.* **2021**, *70*, 7322–7334. [CrossRef]
13. Tagliaferri, D.; Rizzi, M.; Nicoli, M.; Tebaldini, S.; Russo, I.; Monti-Guarnieri, A.V.; Prati, C.M.; Spagnolini, U. Navigation-Aided Automotive SAR for High-Resolution Imaging of Driving Environments. *IEEE Access* **2021**, *9*, 35599–35615. [CrossRef]
14. Manzoni, M.; Tagliaferri, D.; Rizzi, M.; Tebaldini, S.; Monti-Guarnieri, A.V.; Prati, C.M.; Nicoli, M.; Russo, I.; Duque, S.; Mazzucco, C.; et al. Motion Estimation and Compensation in Automotive MIMO SAR. *arXiv* **2022**, arXiv:2201.10504.
15. Cumming, I.G.; Wong, F.H.c. *Digital Processing of Synthetic Aperture Radar Data: Algorithms and Implementation*; Artech House Remote Sensing Library, Artech House: Boston, MA, USA, 2005;
16. Meta, A.; Hoogeboom, P.; Ligthart, L.P. Signal Processing for FMCW SAR. *IEEE Trans. Geosci. Remote Sens.* **2007**, *45*, 3519–3532. [CrossRef]
17. Ulander, L.; Hellsten, H.; Stenstrom, G. Synthetic-aperture radar processing using fast factorized back-projection. *IEEE Trans. Aerosp. Electron. Syst.* **2003**, *39*, 760–776. [CrossRef]
18. Frey, O.; Magnard, C.; Ruegg, M.; Meier, E. Focusing of Airborne Synthetic Aperture Radar Data From Highly Nonlinear Flight Tracks. *IEEE Trans. Geosci. Remote Sens.* **2009**, *47*, 1844–1858. [CrossRef]
19. Tebaldini, S.; Rocca, F.; Mariotti d'Alessandro, M.; Ferro-Famil, L. Phase Calibration of Airborne Tomographic SAR Data via Phase Center Double Localization. *IEEE Trans. Geosci. Remote Sens.* **2016**, *54*, 1775–1792. [CrossRef]

20. Pinto, M.A.; Fohanno, F.; Tremois, O.; Guyonic, S. Autofocusing a synthetic aperture sonar using the temporal and spatial coherence of seafloor reverberation. In Proceedings of the High Frequency Seafloor Acoustics (SACLANTCEN Conference Proceedings CP-45), Lerici, Italy, 30 June–4 July 1997; pp. 417–424.
21. Prats, P.; de Macedo, K.A.C.; Reigber, A.; Scheiber, R.; Mallorqui, J.J. Comparison of Topography- and Aperture-Dependent Motion Compensation Algorithms for Airborne SAR. *IEEE Geosci. Remote Sens. Lett.* **2007**, *4*, 349–353. [CrossRef]
22. Tebaldini, S.; Rizzi, M.; Manzoni, M.; Guarnieri, A.M.; Prati, C.; Tagliaferri, D.; Nicoli, M.; Spagnolini, U.; Russo, I.; Mazzucco, C.; et al. A Quick and Dirty processor for automotive forward SAR imaging. In Proceedings of the 2022 IEEE Radar Conference (RadarConf22), New York City, NY, USA, 21–25 March 2022; pp. 1–6. [CrossRef]