sensors

MDPI

Article

Improving the Accuracy of Two-Color Multiview (2CMV) Advanced Geospatial Information (AGI) Products Using Unsupervised Feature Learning and Optical Flow

Berkay Kanberoglu [1,*] and David Frakes [2]

[1] School of Electrical, Computer and Energy Engineering, Arizona State University, Tempe, AZ 85281, USA
[2] School of Biological and Health Systems Engineering, Arizona State University, Tempe, AZ 85281, USA; dfrakes@asu.edu
* Correspondence: bkanbero@asu.edu

Received: 5 May 2019; Accepted: 3 June 2019; Published: 8 June 2019

Abstract: In two-color multiview (2CMV) advanced geospatial information (AGI) products, temporal changes in synthetic aperture radar (SAR) images acquired at different times are detected, colorized, and overlaid on an initial image such that new features are represented in cyan, and features that have disappeared are represented in red. Accurate detection of temporal changes in 2CMV AGI products can be challenging because of 'speckle noise' susceptibility and false positives that result from small orientation differences between objects imaged at different times. Accordingly, 2CMV products are often dominated by colored pixels when changes are detected via simple pixel-wise cross-correlation. The state-of-the-art in SAR image processing demonstrates that generating efficient 2CMV products, while accounting for the aforementioned problem cases, has not been well addressed. We propose a methodology to address the aforementioned two problem cases. Before detecting temporal changes, speckle and smoothing filters mitigate the effects of speckle noise. To detect temporal changes, we propose using unsupervised feature learning algorithms in conjunction with optical flow algorithms that track the motion of objects across time in small regions of interest. The proposed framework for distinguishing between actual motion and misregistration can lead to more accurate and meaningful change detection and improve object extraction from an SAR AGI product.

Keywords: SAR; 2CMV; change detection; optical flow; k-means; K-SVD

1. Introduction

One important use of synthetic aperture radar (SAR) imagery is in detecting changes between datasets from different imaging passes. Target and coherent change detection in SAR images have been extensively researched [1–4]. In two-color multiview (2CMV) advanced geospatial information (AGI) products, the changes are colorized and overlaid on an initial image such that new features are represented in cyan, and features that have disappeared are represented in red. In order to create the change maps, images are cross-correlated pixel-by-pixel to detect the changes. 2CMV products show changes at the pixel level and are often misleadingly dominated with red and cyan colors. Figure 1 shows a portion of a sample 2CMV image. In the sample images, there is an airplane visibly parked next to a building near the bottom center. It can be seen that many of the pixels in the 2CMV image are colored either red or cyan even if there is no change in the area.

Useful interpretation of temporal changes represented in 2CMV AGI products can be challenging because of speckle noise susceptibility and false positives that result from small orientation differences between objects imaged at different times. When every small intensity change creates a colored pixel, it becomes more difficult for operators and/or algorithms to detect meaningful changes and identify corresponding objects of interest.

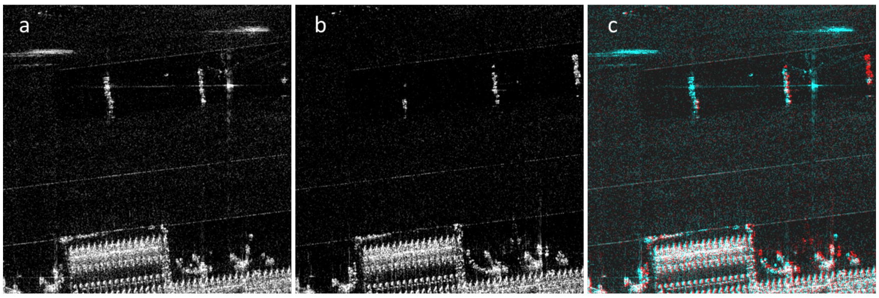

Figure 1. (**a**) reference image; (**b**) mission image; (**c**) two-color multiview (2CMV) image. In both images, there is an airplane visibly parked next to an airport building near the bottom center. In the second image (**b**), the airplane seems rotated by a small degree. The sharp edges of the building are slightly misregistered in the images and these registration errors are false positives in the 2CMV image.

In this work, we introduce a new framework of image processing methods for the efficient generation of 2CMV products toward extraction of advanced geospatial intelligence. Before false positive and object detection algorithms are performed, speckle and smoothing filters are used to mitigate the effects of speckle noise. Then, the number of false positive detections is reduced by applying: (1) unsupervised feature learning algorithms and (2) optical flow algorithms that track the motion of objects across time in small regions of interest.

There have been a number of change detection studies using thresholding [5–8], extreme learning machine [9,10], Markov random fields [11,12] and combinations of feature learning and clustering algorithms [13–19]. Optical flow fields can be used to distinguish between objects that have actually moved between frames and those that are in the same location but are slightly misregistered. Both cases of apparent motion can result in 2CMV detection, but they obviously differ greatly in terms of meaning. Investigation of the state-of-the-art in SAR image processing indicates that differentiating between these two general cases is a problem that has not been well addressed. Algorithms that mitigate speckle noise effects well and distinguishing between actual motion and misregistration can lead to better change detection. There is a lack of published methods for efficient generation of 2CMV products from SAR images, which serves as another motivating factor for this work.

The paper is organized in four sections. Following this introduction, Section 2 gives a brief background on the filtering, unsupervised feature learning, and optical flow techniques that were used and describes the stages of the proposed framework. Section 3 presents simulation results. Section 4 discusses the results and the contributions of the proposed methods.

2. Materials and Methods

In this section, we describe the key methods and steps of our image processing approach for generating change maps that drive the 2CMV representation and eliminating false positives in those maps.

2.1. Speckle Noise Filtering

Speckle noise is an inherent problem in SAR images [20] and causes difficulties for image interpretation by increasing the mean grey level of a local region. In order to mitigate speckle noise effects, we tested different speckle filter designs. Filters that were included in the testing were Frost [21], Enhanced Frost [22], Lee [23], Gamma-MAP [24], SRAD [25] and Non-Local Means [26]. In the end, Enhanced Frost filter was used in the algorithm due to its relatively straightforward implementation and comparable performance.

In [22], it was proposed to divide images into areas of three classes. The first class is comprised of homogeneous areas. The second class is comprised of heterogeneous areas wherein speckle noise is to be reduced, while preserving texture. The third class is comprised of areas containing isolated point targets that filtering should preserve. The Enhanced Frost filter output can be given as:

$$\hat{I}(t_o) = \begin{cases} \bar{I}, & \text{for } C_l(t_o) < C_u, \\ IK_1 \exp\left[-K(C_l(t_o) - C_u)/(C_{max} - C_l(t_o))|t|\right], & \text{for } C_u \le C_l(t_o) \le C_{max}, \\ I & \text{for } C_l(t_o) \ge C_{max}, \end{cases} \tag{1}$$

where $t_o = (x_o, y_o)$ is the spatial coordinate, $\bar{I}$ is the mean intensity value inside the kernel, K is the filter parameter, K_1 is a normalizing constant, and $|t|$ is the absolute value of the pixel distance from the center of the kernel at t_o. The rest of the parameters are

$$C_u = \sqrt{\frac{1}{L}},$$
$$C_l(t_o) = \sigma/\bar{I}, \text{ and}$$
$$C_{max} = \sqrt{1 + \frac{2}{L}},$$

where C_u is the speckle coefficient of variation of the image, $C_l(t_o)$ is the local coefficient of variation of the filter kernel centered at t_o, C_{max} is the upper speckle coefficient of variation of the image, and L is the number of looks. In our implementation, instead of L, we used "equivalent number of looks" (ENL). It can be defined as $ENL = \mu^2/\sigma^2$, where μ is the mean and σ is the standard deviation.

2.2. *k*-Means Clustering

The k-means clustering algorithm attempts to partition p observations into k clusters such that each observation belongs to the nearest cluster mean (centroid) [27]. The k-means algorithm iteratively tries to find k centroids for each cluster, while minimizing a within-cluster sum of squares

$$argmin \sum_{i=1}^{k} \sum_{x_j \epsilon S} \|x_j - \mu_j\|^2,$$

where x_j is the j^{th} observation and μ_j is the mean point (centroid) in the cluster. The basic steps of the algorithm are given in Algorithm 1:

Algorithm 1 *k*-means clustering algorithm

```
1. Initialize the centroids: Assign $k$ points as the initial group centroids.
2. Calculate the distance of each point to the centroids and assign the point to the cluster that has the closest centroid.
3. After the assignment of all the points, recalculate the new values of the centroids.
4. Repeat Steps 2 and 3 until the centroid locations converge to a fixed value.
```

2.3. K-SVD

K-SVD is a dictionary learning algorithm that is used for training overcomplete dictionaries for sparse representations of signals [28,29]. It is an iterative method that is a generalization of the k-means clustering algorithm. The K-SVD algorithm alternates between two stages: (1) sparse coding stage, and (2) dictionary update stage. In the first stage, a pursuit algorithm is used to sparsely code the input data based on the current dictionary. Based on Ref. [29], the Batch Orthogonal Matching Pursuit (Batch-OMP) algorithm can be used in this step. In the second stage, the dictionary atoms are updated to better fit the data via a singular value decomposition (SVD) approach. The basic steps of the K-SVD algorithm are given in Algorithm 2.

Algorithm 2 K-SVD algorithm

```
Task: Find the best dictionary to represent the data samples $\{y_i\}_{i=1}^{N}$, $y_i \epsilon R^N$ as sparse compositions by solving:
    $min_{D,X}\{\|Y - DX\|_F^2\}$ subject to $\forall_i$, $\|x_i\|_0 \leq T_0$.
Initialization: Set the dictionary matrix $D^{(0)}\ \epsilon\ R^{n \times K}$ with $l^2$ normalized columns. Set $J = 1$.
Iterations: Repeat until convergence:
  • Sparse coding stage: Use any pursuit algorithm to compute the representation vectors $x_i$ for each sample $y_i$ by approximating the solution of
        $i = 1, 2, ..., N, \quad min_{x_i}\{\|y_i - Dx_i\|_2^2\}$ subject to $\|x_i\|_0 \leq T_0$.
  • Dictionary update stage: For each column $k = 1, 2, ..., K$ in $D^{J-1}$,
      – Define the group of samples that use this atom, $w_k = \{i\ |1 \leq i \leq N, x_T^k(i) \neq 0\}$
      – Compute the overall representation error matrix, $E_k$, by
            $E_k = Y - \sum_{j \neq k} d_j x_T^j$
      – Restrict $E_k$ by choosing only the columns corresponding to $w_k$, and obtain $E_k^R$.
      – Apply SVD decomposition $E_k^R = U \Delta V^T$. Choose the updated dictionary column $\widetilde{d_k}$ to be the first column of $U$.
        Update the coefficient vector $x_R^k$ to be the first column of $V$ multiplied by $\Delta(1, 1)$.
  • Set $J = J + 1$.
```

2.4. Optical Flow

Optical flow is the apparent motion of objects in image sequences that results from relative motion between the objects and the imaging perspective. In one canonical optical flow paper [30], two kinds of constraints are introduced in order to estimate the optical flow: the smoothness constraint and the brightness constancy constraint. In this section, we give a brief overview of the optical flow algorithm we employ in the proposed methodology.

Optical flow methods estimate the motion between two consecutive image frames that were acquired at times t and $t + \delta t$. A flow vector for every pixel is calculated. The vectors represent approximations of image motion that are based in large part on local spatial derivatives. Since the flow velocity has two components, two constraints are needed to solve for it. The brightness constancy constraint assumes that the brightness of a small area in the image remains constant as the area moves from image to image. Image brightness at the point (x,y) in the image at time t is denoted here as $I(x, y, t)$. If the point moves by δx and δy in time δt, then, according to the brightness constancy constraint:

$$\frac{dI}{dt} = 0. \tag{2}$$

This can also be stated as:

$$I(\mathbf{r} + \delta\mathbf{r}, t + \delta t) = I(\mathbf{r}, t), \tag{3}$$

where $\mathbf{r} = (x, y, 1)^T$ and $\mathbf{r} + \delta\mathbf{r} = (x + \delta x, y + \delta y, 1)^T$. However, the brightness constancy constraint is restrictive. A less restrictive brightness constraint was chosen to address the intensity changes in SAR images. In Reference [31], it is proposed that the brightness constancy constraint can be replaced with a more general constraint that allows a linear transformation between the pixel brightness values. This way, the brightness change can be non-zero, or:

$$\frac{dI}{dt} \neq 0.$$

The formulation that allows a linear transformation between the pixel brightness values is less restrictive, and can be written as:

$$I(\mathbf{r} + \delta\mathbf{r}, t + \delta t) = M(\mathbf{r}, t) I(\mathbf{r}, t) + C(\mathbf{r}, t). \tag{4}$$

After using the Taylor series, the revised constraint equation can be obtained:

$$I_t + I_{\mathbf{r}} \cdot \mathbf{r}_t - I m_t - c_t = 0, \tag{5}$$

where $m_t = \lim_{\delta t \to 0} \frac{\delta m}{\delta t}$ and $c_t = \lim_{\delta t \to 0} \frac{\delta c}{\delta t}$.

The relaxed brightness constraint error is:

$$\epsilon_I = \iint (I_t + I_{\mathbf{r}} \cdot \mathbf{r}_t - I m_t - c_t)^2 \, dx \, dy. \tag{6}$$

Equation (6) can be combined with the other constraint errors to produce the final functional to be minimized:

$$\epsilon_{total} = \epsilon_I + \lambda_s \epsilon_s + \lambda_m \epsilon_m + \lambda_c \epsilon_c, \tag{7}$$

where λ_s, λ_m, and λ_c are error weighting coefficients. The remaining errors are given as:

$$\begin{aligned} \epsilon_s &= \iint ||\nabla \mathbf{r}_t||_2^2 dxdy, \\ \epsilon_m &= \iint ||\nabla m_t||_2^2 dxdy, \\ \epsilon_c &= \iint ||\nabla c_t||_2^2 dxdy. \end{aligned}$$

Substituting the approximated Laplacians into the Euler–Lagrange equations, a single matrix equation can be derived:

$$\mathbf{A}\mathbf{f} = \mathbf{g}(\bar{\mathbf{f}}), \tag{8}$$

where

$$\mathbf{A} = \begin{pmatrix} I_x^2 + \lambda_s & I_x I_y & -I_x I & -I_x \\ I_x I_y & I_y^2 + \lambda_s & -I_y I & -I_y \\ -I_x I & -I_y I & I^2 + \lambda_m & I \\ -I_x & -I_y & I & 1 + \lambda_c \end{pmatrix}, \mathbf{f} = \begin{pmatrix} u \\ v \\ m_t \\ c_t \end{pmatrix}, \ \mathbf{g}(\bar{\mathbf{f}}) = \begin{pmatrix} \lambda_s \bar{u} - I_x I_t \\ \lambda_s \bar{v} - I_y I_t \\ \lambda_m \bar{m}_t + I_t I \\ \lambda_c \bar{c}_t + I_t \end{pmatrix}.$$

These equations have to be solved iteratively. The solution is given by:

$$\mathbf{f} = \mathbf{A}^{-\mathbf{1}} \mathbf{g}(\bar{\mathbf{f}}), \tag{9}$$

where

$$\mathbf{A}^{-\mathbf{1}} = \frac{1}{\alpha} \begin{pmatrix} \begin{matrix} \lambda_c \lambda_m \lambda_s + \lambda_m \lambda_s + \\ I^2 \lambda_c \lambda_s + I_y^2 \lambda_c \lambda_m \end{matrix} & -I_x I_y \lambda_c \lambda_m & I_x I \lambda_c \lambda_s & I_x \lambda_m \lambda_s \\ -I_x I_y \lambda_c \lambda_m & \begin{matrix} \lambda_c \lambda_m \lambda_s + \lambda_m \lambda_s + \\ I^2 \lambda_c \lambda_s + I_y^2 \lambda_c \lambda_m \end{matrix} & I_y I \lambda_c \lambda_s & I_y \lambda_m \lambda_s \\ -I_x I \lambda_c \lambda_s & I_y I \lambda_c \lambda_s & \begin{matrix} (I_x^2 + I_y^2) \lambda_c \lambda_s + \\ \lambda_c \lambda_s^2 + \lambda_s^2 \end{matrix} & -I \lambda_s^2 \\ I_x \lambda_m \lambda_s & I_y \lambda_m \lambda_s & -I \lambda_s^2 & \begin{matrix} (I_x^2 + I_y^2) \lambda_m \lambda_s + \\ \lambda_m \lambda_s^2 + I^2 \lambda_s^2 \end{matrix} \end{pmatrix}$$

and

$$\alpha = \lambda_m \lambda_s^2 + I^2 \lambda_c \lambda_s^2 + (I_x^2 + I_y^2 + \lambda_s) \lambda_c \lambda_m \lambda_s.$$

The equations can then be solved iteratively for other pixels with:

$$\mathbf{f}^{\mathbf{k}+\mathbf{1}} = \mathbf{A}^{-\mathbf{1}}\mathbf{g}(\bar{\mathbf{f}}^{\mathbf{k}}), \tag{10}$$

where k is the iteration number. This way the matrix $\mathbf{A}^{-\mathbf{1}}$ need only be computed once. More details about this optical flow algorithm can be found in Ref. [31].

2.5. Image Processing Steps

In this section, we describe the image processing approach for extracting change maps. The inputs are two registered SAR images of the same field of view that were taken at different times, i.e., "reference" image and "mission" image. Due to the large size of the images, images were divided into subimages for processing.

In the denoising step, an Enhanced Frost filter, as described in Section 2.1, with a 5 × 5 window size was first used to mitigate the speckle noise effects. Then, a 9 × 9 low pass filter was used to smooth the test areas in order to obtain more uniform flow fields in the optical flow processing step. The remaining steps are grouped in three stages and described in the following subsections. The detailed flow diagram shown in Figure 2 can be used as a guide for the following descriptions.

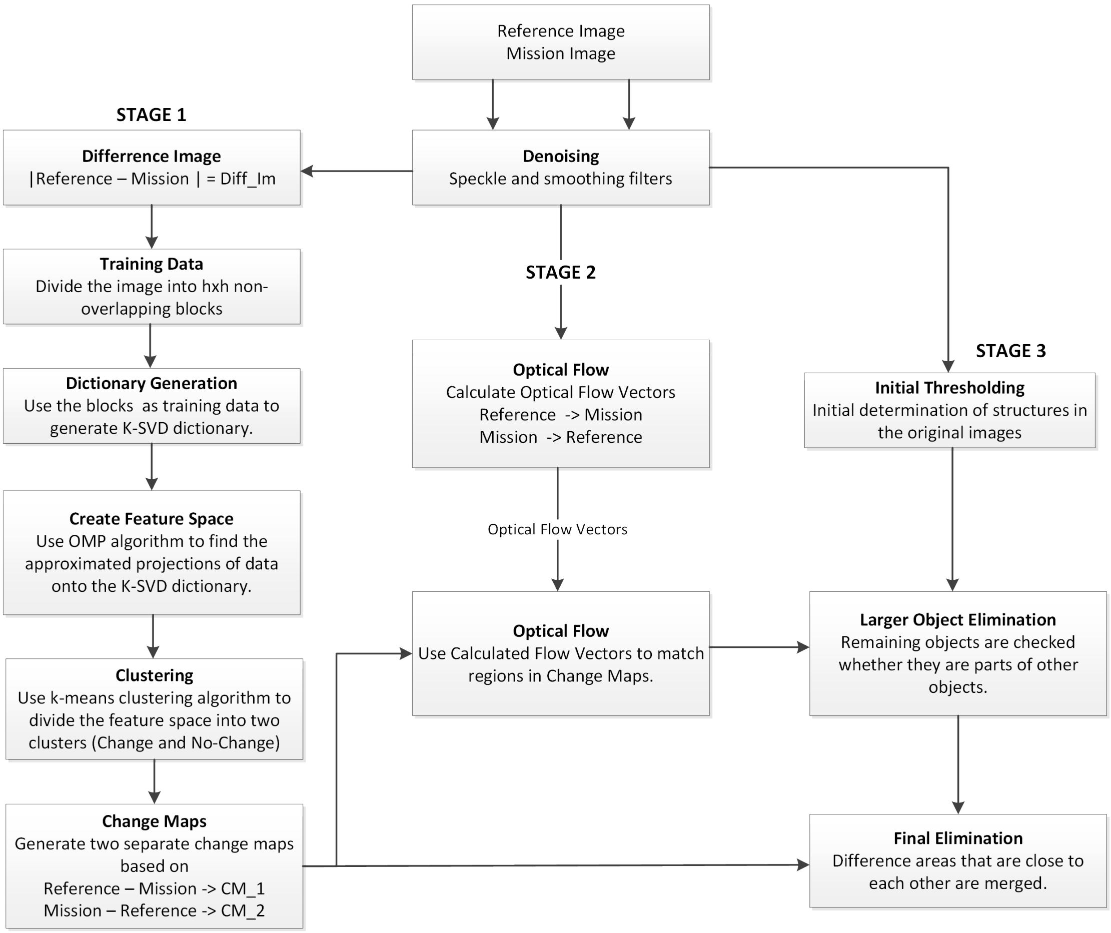
Reference Image
Mission Image
STAGE 1
Difference Image
|Reference – Mission | = Diff_Im
Denoising
Speckle and smoothing filters
Training Data
Divide the image into hxh non-overlapping blocks
STAGE 2
Dictionary Generation
Use the blocks as training data to generate K-SVD dictionary.
Optical Flow
Calculate Optical Flow Vectors
Reference -> Mission
Mission -> Reference
STAGE 3
Initial Thresholding
Initial determination of structures in the original images
Create Feature Space
Use OMP algorithm to find the approximated projections of data onto the K-SVD dictionary.
Optical Flow Vectors
Optical Flow
Use Calculated Flow Vectors to match regions in Change Maps.
Larger Object Elimination
Remaining objects are checked whether they are parts of other objects.
Clustering
Use k-means clustering algorithm to divide the feature space into two clusters (Change and No-Change)
Change Maps
Generate two separate change maps based on
Reference – Mission -> CM_1
Mission – Reference -> CM_2
Final Elimination
Difference areas that are close to each other are merged.

Figure 2. Flow diagram of the proposed framework.

2.5.1. First Stage: Generation of Change Maps Using Unsupervised Feature Learning

Two change maps are needed for a 2CMV representation of an SAR image pair. Each change map represents the changes that exist in the corresponding SAR image. In this stage, we generate a combined change map and separate it into two change maps. In order to generate the combined change map, we used an approach similar to that was used in [13]. In the original approach, an eigenvector space is created by performing principle component analysis (PCA) on the difference image and k-means algorithm classifies the projections onto the eigenvector space into two classes: e.g., change and no-change. The basic steps are given in Algorithm 3. It should be noted that, in our framework, PCA was replaced with K-SVD because one can adjust the dictionary size and the sparsity constraint to obtain change maps with different levels of details. Figure 3 shows two change map results with different dictionary sizes.

Algorithm 3 Generating change maps

- **Difference Image:**

$$X_{dif} = |Reference - Mission|$$

- **Training Data:** Divide X_{dif} into hxh non-overlapping blocks.
- **Dictionary Generation:** Use the K-SVD algorithm to generate an overcomplete dictionary.
- **Create Feature Space:**
  - Generate hxh blocks for each pixel in X_{dif} where the pixel is in the center of the block.
  - Use OMP algorithm to generate the projections of the data onto the dictionary.
- **Clustering:** Use the k-means algorithm to classify the feature space into two classes, e.g. change and no-change.
- **Change maps:** Use the two classes to generate the combined change map. Divide the combined change map into two separate change maps based on the changes that occur in the images.

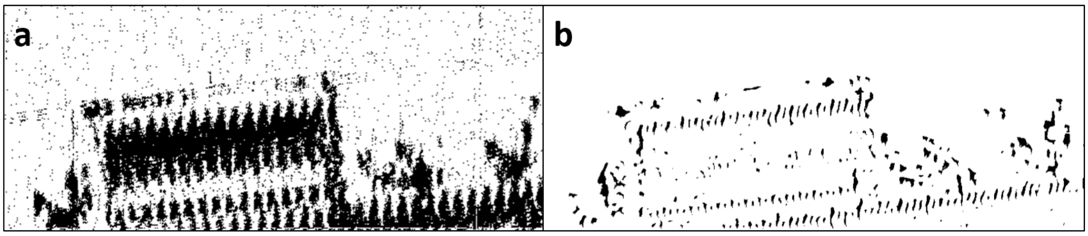

Figure 3. (**a**) change map with dictionary size = 30 atoms with 30 non-zero coefficients; (**b**) change map with dictionary size = 15 with three non-zero coefficients. Note that a larger dictionary size with more non-zero coefficients captures more changes.

After the change maps are generated, object properties such as area and location are calculated and, based on a user-defined area threshold, insignificant change areas are excluded from the change maps. The remaining change areas are then overlaid onto the reference image. In the 2CMV image, the areas that exist only in the reference image are colored in cyan and the areas that exist only in the mission image are colored in red. A sample 2CMV image after this stage is shown in Figure 4.

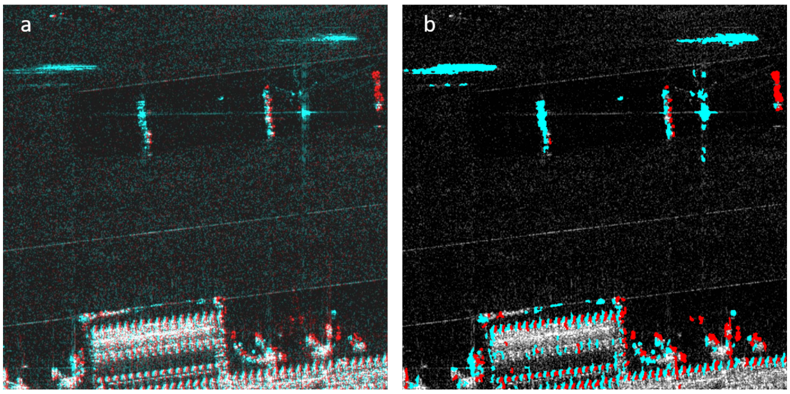

Figure 4. (**a**) original 2CMV image; (**b**) 2CMV image after Stage 1. Note that there are several false positives around the ridges of the building. In the second image, change colors (red and cyan) were made more pronounced to highlight the false positives.

In a previous work, this stage was replaced by adaptive thresholding [32].

2.5.2. Second Stage: Optical Flow

Figure 4 displays a 2CMV image after the first stage wherein it is clear that additional processing is needed to improve results because the ridges of the building in both images are slightly misregistered and they are shown as changes in both images. The primary improvement that is targeted with additional processing is reducing the number of false positives in the image. This goal can be accomplished with the use of the optical flow (OF) method described in Section 2.4. To manage computational complexity, the optical flow algorithm is performed on 256 × 256 pixel image blocks. Note that optical flow is calculated based on the original reference and mission images.

After obtaining the flow vectors, the direction of the majority of flow vectors is determined. The flow vectors that are in this direction are applied to the two first stage change maps to find matches. In the reference image, OF vectors are used to move the detected change areas in the flow direction. The destination of an area is then compared with the same location in the mission image. If there is a matching area based on location and size, then the two change areas are excluded from the change maps. The same process is performed in the opposite direction to match mission image change areas in the reference image. Figure 5 illustrates this step.

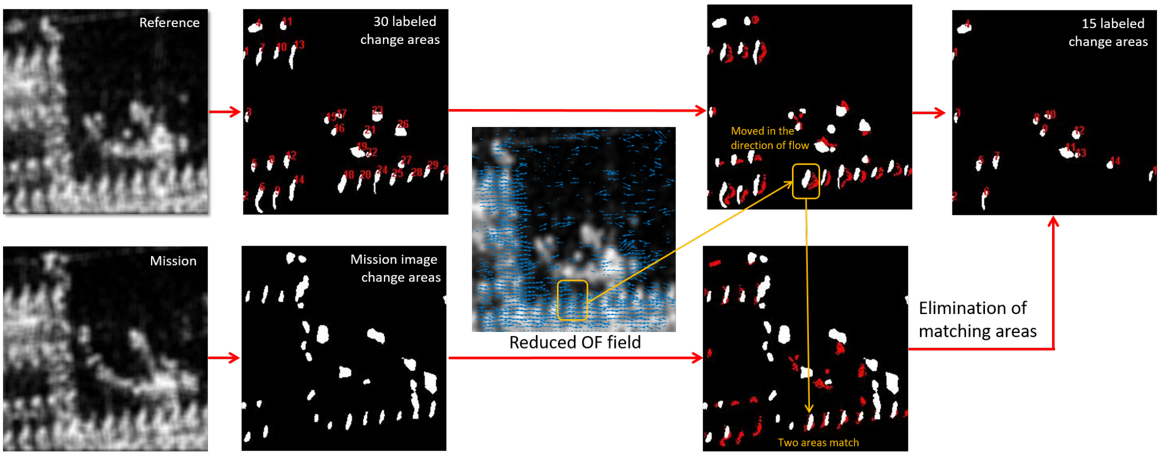

Figure 5. Elimination of false positives using optical flow. Change areas are moved along the flow direction in the reference image change map. Moved areas (shown in red) from the reference image are overlaid onto the mission image change map. The overlapped areas are then removed.

2.5.3. Third Stage: OF Assisted Object Extraction

This stage has two main parts: extraction and elimination. Extraction is performed by an adaptive thresholding method that is similar to the one used in [32]. In this stage, the thresholding is performed on the original images to extract/label objects. The resulting two thresholded images are processed in two ways. First, OF vectors are used on the images to match the objects. The main difference from the second stage is that the flow vectors are used on the original thresholded images, not on the change maps. Change maps do not necessarily contain objects, and the goal is to find objects that moved between the two images. Objects with possibility of movement are labeled and compared against the areas in the change maps. It should also be noted that only some parts of an object can be detected as a change, and these detected changes can be used as a guide to extract the full object.

After this process, the labeled areas in the change maps are overlaid on the reference image and checked whether they are a part of a larger object in the image. If the labeled area is found to be a part of a larger object, then the same location in the mission image is checked for the same object. In the case of two similar objects around the same location, it can be assumed that the detected object is a false negative and excluded from the difference map. After these two methods are performed, the output of this stage is generated by simply taking the intersection of the two results. Figure 6 shows how this process converts the reference image in (a) to the final output in (e).

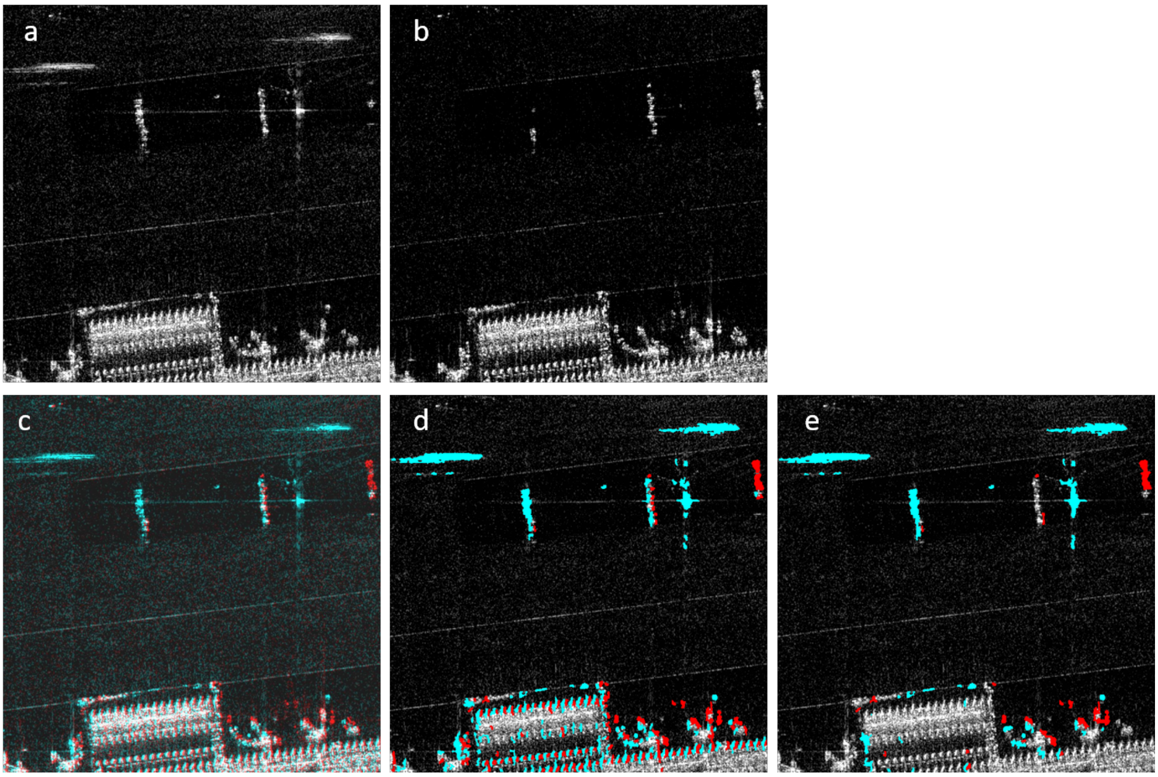

Figure 6. (**a**) reference image; (**b**) mission image; (**c**) original 2CMV image; (**d**) 2CMV image after using dictionary learning and clustering (Stage 1); (**e**) final 2CMV image. False positives are reduced.

3. Results

The proposed algorithm was compared against three change detection methods: PCAKM [13], GaborTLC [18], and NR-ELM [10]. All three methods are implemented with their default parameters by using the publicly available code provided by the authors. The first dataset consisted of 1024 × 1024 regions from an SAR image pair provided by Lockheed Martin (Bethesda, MD, USA). The data were acquired with various Lockheed Martin SAR units, one example of which is an airborne long range, all weather, day/night, X-band SAR unit with a resolution of 1 m. The selected regions contained speckle noise and false positives that resulted from registration and perspective problems. 2CMV images were generated for each method. The visual results are shown in Figure 7. NR-ELM was more susceptible to noise compared to the other methods. It was noted that unsupervised dictionary learning and clustering algorithms were effective at removing false positives that did not match object profiles. Optical flow was effective for removing difficult false positives that resulted from registration and perspective problems.

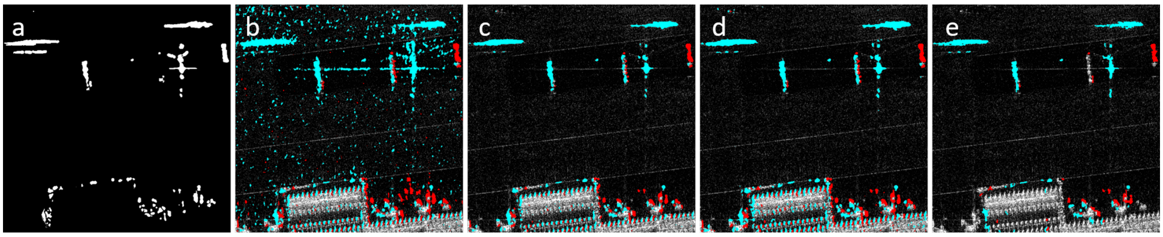

Figure 7. Results by (**a**) manual ground truth; (**b**) NR-ELM; (**c**) GaborTLC; (**d**) PCAKM; (**e**) proposed method.

From the ground truth map, the actual number of pixels belonging to the unchanged class and changed class are calculated, denoted as N_u and N_c, respectively. With this information, five objective metrics are adopted for quantitative evaluation. False positive (FP) is the number of pixels belonging to the unchanged class but falsely classified as changed class. False negative (FN) is the number of pixels belonging to the changed class but falsely classified as unchanged class. The overall error (OE) is calculated by FP + FN. Percentage correct classification (PCC) and Kappa coefficient (KC) are as follows:

$$PCC = \frac{(N_c - FN) + (N_u - FP)}{N_u + N_c} \times 100\%,$$

$$KC = \frac{PCC - PRE}{1 - PRE},$$

where proportional reduction in error (PRE) is defined as

$$PRE = \frac{(N_c - FN + FP) \cdot N_c + (N_u - FP + FN) \cdot N_u}{(N_u + N_c)^2}.$$

The results of the quantitative metrics are given in Table 1.

Table 1. Results for the SAR dataset.

Methods	FP	FN	OE	PCC(%)	KC(%)	Time(s)
NR-ELM	97377	4276	101653	0.9031	0.2993	202.1
GaborTLC	20160	8449	28609	0.9727	0.5809	74.6
PCAKM	35135	6251	41386	0.9605	0.51	6.4
Proposed method	3865	12852	16717	0.9841	0.6569	199.1

In addition to these results, the proposed framework was tested on an ensemble of 1024 × 1024 regions from the same SAR dataset. In many representative image regions where registration errors were prevalent, false positive detections were reduced by over 60%. Filtering of speckle noise and adaptive thresholds improved the quality of the object extraction and helped identify false positives. Establishing false positive motion/error thresholds, in accordance with initial image registration, can be key for continued improvement. It is also a challenge to extract only regions with intensity value changes. It is possible that wavelet based methods might be more successful with such a task.

For the second test, a more standard dataset was used. The San Francisco dataset has been used in change detection studies and its ground truth change map was provided in [33]. It consists of two SAR images over the city of San Francisco that were acquired by ERS-2 C-band SAR sensor with VV polarization. The images were provided by the European Space Agency with a resolution of 25-m. These two images were captured in August 2003 and May 2004, respectively. The size of the images were 256 × 256 for this test. The change maps of the methods can be seen in Figure 8.

The results of the quantitative metrics are given in Table 2. The proposed framework performed comparable to PCAKM as a change detection algorithm. The San Francisco dataset doesn't contain registration and perspective errors with speckle noise.

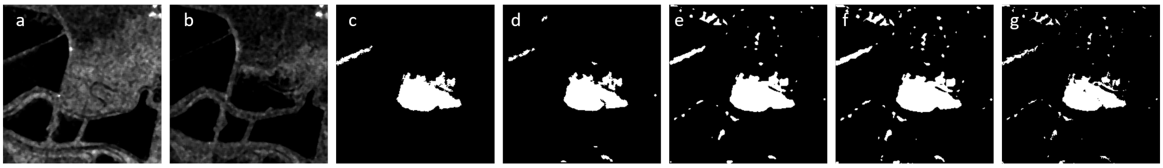

Figure 8. (**a**,**b**) San Francisco dataset; (**c**) ground truth; (**d**) NR-ELM; (**e**) GaborTLC; (**f**) PCAKM; (**g**) proposed method.

Table 2. Results for the San Francisco dataset

Methods	FP	FN	OE	PCC(%)	KC(%)
NR-ELM	328	440	768	0.9883	0.9107
GaborTLC	1376	60	1436	0.9781	0.8539
PCAKM	1855	73	1928	0.9706	0.8115
Proposed method	836	685	1521	0.9768	0.8277

It should be noted that the proposed framework provided better results compared to the other methods when the datasets contain registration and perspective errors with speckle noise. Otherwise, the performance of the proposed method is comparable to PCAKM as a change detection algorithm since the optical flow processing stage cannot provide matching regions in the images.

Even though the computational complexity was not an issue during the course of this work, the speckle filtering, optical flow processing and merging are computationally expensive processes. On a dual core computer (Intel Core i7 6500U, Santa Clara, CA, USA) with 16 GB of memory, it takes slightly less than 3.5 min to process one region. There are many factors that are contributing to this time. Code was written in the MATLAB environment (R2016a, MathWorks, Natick, MA, USA) and not optimized for performance.

4. Conclusions

It was shown that unsupervised feature learning algorithms can be effectively used in conjunction with optical flow methods to generate 2CMV AGI products. Other image processing methods like noise reduction and adaptive thresholding were used to improve object extraction in the proposed methodology. Results demonstrated the ability of the techniques to reduce false positives by up to 60% in the provided SAR image pairs. However, there is still room for further improvement. For example, it was noticed that optical flow object matches close to image block borders can be overlooked due to the inaccuracy of flow vectors near the block borders. This problem can be addressed with a multigrid approach that leverages overlapping image blocks. Using this approach, if an object pair is close to the border in one block, then it will be near the center of an overlapping block. It has also been noted that only some parts of an object can be detected as a change, and the detected parts can be used as a guide to segment the full object. Objects that are close to one another can be merged to provide a more holistic analysis of the scene and further reduce the number of false positive object detections. However, it must be concurrently ensured that false positive reduction is not overly aggressive to the point that false negatives are generated. More recent optical flow or motion estimation algorithms can be investigated as an alternative to the one utilized in this work. The chosen optical flow method is suitable for the tested dataset and performs adequately as expected since it takes into account the intensity changes between images. The choice of K-SVD over PCA increased the computational complexity while allowing flexibility over the details of the change maps by changing the dictionary size and the number of non-zero coefficients. Dictionaries with higher number of non-zero coefficients provided more detailed change maps. For future work, investigating the correlation between the quantitative metrics and the parameters in the framework (e.g., dictionary size, etc.) can provide insight into tuning the framework for different types of datasets. Other methods can be researched as alternatives to the K-SVD method in the framework.

Author Contributions: Conceptualization, B.K. and D.F.; methodology, B.K.; software, B.K.; validation, B.K. and D.F.; investigation, B.K.; data curation, D.F.; writing—original draft preparation, B.K.; writing—review and editing, D.F.; supervision, D.F.; project administration, D.F.; funding acquisition, D.F.

Funding: This work was supported in part by Lockheed Martin.

Acknowledgments: The authors would like to acknowledge SenSIP for the center's valuable contributions to this work.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. El-Darymli, K.; McGuire, P.; Power, D.; Moloney, C. Target Detection in Synthetic Aperture Radar Imagery: A State-of-the-Art Survey. *J. Appl. Remote Sens.* **2013**, *7*, 071598.
2. El-Darymli, K.; Gill, E.W.; McGuire, P.; Power, D.; Moloney, C. Automatic Target Detection in Synthetic Aperture Radar Imagery: A State-of-the-Art Review. *IEEE Access* **2016**, *4*, 6014–6058.
3. Ashok, H.G.; Patil, D.R. Survey on Change Detection in SAR Images. In Proceedings of the IJCA Proceedings on National Conference on Emerging Trends in Computer Technology, Shirpur, India, 28–29 March 2014; pp. 4–7.
4. Ren, W.; Song, J.; Tian, S.; Wu, W. Survey on Unsupervised Change Detection Techniques in SAR Images1. In Proceedings of the 2014 IEEE China Summit International Conference on Signal and Information Processing (ChinaSIP), Xi'an, China, 9–13 July 2014; pp. 143–147.
5. Bazi, Y.; Bruzzone, L.; Melgani, F. An Unsupervised Approach Based on the Generalized Gaussian Model to Automatic Change Detection in Multitemporal SAR Images. *IEEE Trans. Geosci. Remote Sens.* **2005**, *43*, 874–887.
6. Bovolo, F.; Bruzzone, L. A Detail-Preserving Scale-Driven Approach to Change Detection in Multitemporal SAR Images. *IEEE Trans. Geosci. Remote Sens.* **2005**, *43*, 2963–2972.
7. Moser, G.; Serpico, S.B. Generalized Minimum-Error Thresholding for Unsupervised Change Detection from SAR Amplitude Imagery. *IEEE Trans. Geosci. Remote Sens.* **2006**, *44*, 2972–2982.
8. Sumaiya, M.N.; Kumari, R.S.S. Logarithmic Mean-Based Thresholding for SAR Image Change Detection. *IEEE Geosci. Remote Sens. Lett.* **2016**, *13*, 1726–1728.
9. Jia, L.; Li, M.; Zhang, P.; Wu, Y. SAR Image Change Detection Based on Correlation Kernel and Multistage Extreme Learning Machine. *IEEE Trans. Geosci. Remote Sens.* **2016**, *54*, 5993–6006.
10. Gao, F.; Dong, J.; Li, B.; Xu, Q.; Xie, C. Change Detection from Synthetic Aperture Radar Images Based on Neighborhood-Based Ratio and Extreme Learning Machine. *J. Appl. Remote Sens.* **2016**, *10*, 10–14.
11. Melgani, F.; Bazi, Y. Markovian Fusion Approach to Robust Unsupervised Change Detection in Remotely Sensed Imagery. *IEEE Geosci. Remote Sens. Lett.* **2006**, *3*, 457–461.
12. Yousif, O.; Ban, Y. Improving SAR-Based Urban Change Detection by Combining MAP-MRF Classifier and Nonlocal Means Similarity Weights. *IEEE J. Sel. Top. Appl. Earth Observ.* **2014**, *7*, 4288–4300.
13. Celik, T. Unsupervised Change Detection in Satellite Images Using Principal Component Analysis and k-Means Clustering. *IEEE Geosci. Remote Sens. Lett.* **2009**, *6*, 772–776.
14. Li, W.; Chen, J.; Yang, P.; Sun, H. Multitemporal SAR Images Change Detection Based on Joint Sparse Representation of Pair Dictionaries. In Proceedings of the 2012 IEEE International Geoscience and Remote Sensing Symposium, Munich, Germany, 22–27 July 2012; pp. 6165–6168.
15. Lu, X.; Yuan, Y.; Zheng, X. Joint Dictionary Learning for Multispectral Change Detection. *IEEE Trans. Cybern.* **2017**, *47*, 884–897.
16. Ghosh, A.; Mishra, N.; Ghosh, S. Fuzzy Clustering Algorithms for Unsupervised Change Detection in Remote Sensing Images. *Inf. Sci.* **2011**, *181*, 699–715.
17. Nguyen, L.H.; Tran, T.D. A Sparsity-Driven Joint Image Registration and Change Detection Technique for SAR Imagery. In Proceedings of the 2010 IEEE International Conference on Acoustics, Speech and Signal Processing, Dallas, TX, USA, 14–19 March 2010; pp. 2798–2801.
18. Li, H.; Celik, T.; Longbotham, N.; Emery, W.J. Gabor Feature Based Unsupervised Change Detection of Multitemporal SAR Images Based on Two-Level Clustering. *IEEE Geosci. Remote Sens. Lett.* **2015**, *12*, 2458–2462.
19. Gong, M.; Su, L.; Jia, M.; Chen, W. Fuzzy Clustering With a Modified MRF Energy Function for Change Detection in Synthetic Aperture Radar Images. *IEEE Trans. Fuzzy Syst.* **2014**, *22*, 98–109.
20. Dekker, R.J. Speckle Filtering in Satellite SAR Change Detection Imagery. *Int. J. Remote Sens.* **1998**, *19*, 1133–1146.
21. Frost, V.; Stiles, J.A.; Shanmugan, K.S.; Holtzman, J.C. A Model for Radar Images and Its Application to Adaptive Digital Filtering of Multiplicative Noise. *IEEE Trans. Pattern Anal. Machine Intell.* **1982**, *PAMI-4*, 157–166.
22. Lopes, A.; Touzi, R.; Nezry, E. Adaptive Speckle Filters and Scene Heterogeneity. *IEEE Trans. Geosci. Remote Sens.* **1990**, *28*, 992–1000.

23. Lee, J.S. Digital Image Enhancement and Noise Filtering by Use of Local Statistics. *IEEE Trans. Pattern Anal. Machine Intell.* **1980**, *PAMI-2*, 165–168.
24. Lopes, A.; Nezry, E.; Touzi, R.; Laur, H. Maximum a Posteriori Filtering and First Order Texture Models in SAR Images. In Proceedings of the 10th Annual International Symposium on Geoscience and Remote Sensing, College Park, MD, USA, 20–24 May 1990; pp. 2409–2412.
25. Yu, Y.; Acton, S. Speckle Reducing Anisotropic Diffusion. *IEEE Trans. Image Process.* **2002**, *11*, 1260–1270.
26. Coupe, P.; Hellier, P.; Kervrann, C.; Barillot, C. NonLocal Means-based Speckle Filtering for Ultrasound Images. *IEEE Trans. Image Process.* **2009**, *18*, 2221–2229.
27. Gonzalez, R.; Woods, R. *Digital Image Processing*, 3 ed.; Prentice-Hall: Upper Saddle River, NJ, USA, 2006.
28. Aharon, M.; Elad, M.; Bruckstein, A. K-SVD: An Algorithm for Designing Overcomplete Dictionaries for Sparse Representation. *IEEE Trans. Signal Process.* **2006**, *54*, 4311–4322.
29. Rubinstein, R.; Zibulevsky, M.; Elad, M. *Efficient Implementation of the K-SVD Algorithm Using Batch Orthogonal Matching Pursuit*; Technical Report; Computer Science Department, Technion: Haifa, Israel, 2008.
30. Horn, B.; Schunck, B. Determining Optical Flow. *Artif. Intell.* **1980**, *17*, 185–203.
31. Gennert, M.; Negahdaripour, S. *Relaxing the Brightness Constancy Assumption in Computing Optical Flow*; A.I. Lab Memo 975, Massachusetts Institute of Technology: Cambridge, MA, USA, 1987.
32. Kanberoglu, B.; Frakes, D. *Extraction of Advanced Geospatial Intelligence (AGI) from Commercial Synthetic Aperture Radar Imagery*; International Society for Optics and Photonics: Anaheim, CA, USA, 9–13 April 2017. 2017; Volume 10201, p. 1020106.
33. Gao, F.; Liu, X.; Dong, J.; Zhong, G.; Jian, M. Change Detection in SAR Images Based on Deep Semi-NMF and SVD Networks. *Remote Sens.* **2017**, *9*, 435.