remote sensing

MDPI

Article

Learning Domain-Adaptive Landmark Detection-Based Self-Supervised Video Synchronization for Remote Sensing Panorama

Ling Mei [1], Yizhuo He [2], Farnoosh Javadi Fishani [3], Yaowen Yu [4,*], Lijun Zhang [4] and Helge Rhodin [3]

[1] School of Information Science and Engineering, Wuhan University of Science and Technology, Wuhan 430081, China
[2] School of Computer Science, Carnegie Mellon University (CMU), Pittsburgh, PA 15213, USA
[3] Department of Computer Science, University of British Columbia (UBC), Vancouver, BC V6T 1Z4, Canada
[4] School of Artificial Intelligence and Automation, Huazhong University of Science and Technology, Wuhan 430074, China
* Correspondence: yaowen_yu@hust.edu.cn

Abstract: The synchronization of videos is an essential pre-processing step for multi-view reconstruction such as the image mosaic by UAV remote sensing; it is often solved with hardware solutions in motion capture studios. However, traditional synchronization setups rely on manual interventions or software solutions and only fit for a particular domain of motions. In this paper, we propose a self-supervised video synchronization algorithm that attains high accuracy in diverse scenarios without cumbersome manual intervention. At the core is a motion-based video synchronization algorithm that infers temporal offsets from the trajectories of moving objects in the videos. It is complemented by a self-supervised scene decomposition algorithm that detects common parts and their motion tracks in two or more videos, without requiring any manual positional supervision. We evaluate our approach on three different datasets, including the motion of humans, animals, and simulated objects, and use it to build the view panorama of the remote sensing field. All experiments demonstrate that the proposed location-based synchronization is more effective compared to the state-of-the-art methods, and our self-supervised inference approaches the accuracy of supervised solutions, while being much easier to adapt to a new target domain.

Keywords: video synchronization; remote sensing; image mosaic; self-supervised learning; style transfer

Citation: Mei, L.; He, Y.; Fishani, F.J.; Yu, Y.; Zhang, L.; Rhodin, H. Learning Domain-Adaptive Landmark Detection-Based Self-Supervised Video Synchronization for Remote Sensing Panorama. *Remote Sens.* **2023**, *15*, 953. https://doi.org/10.3390/rs15040953

Academic Editor: Qi Wang and Costas Panagiotakis

Received: 4 November 2022
Revised: 30 January 2023
Accepted: 4 February 2023
Published: 9 February 2023

1. Introduction

Recently, remote sensing image mosaic technology has regained importance in the image processing and pattern recognition community; it can be used for the detection and reconnaissance of Unmanned Aerial Vehicles (UAV), e.g., UAV panoramic imaging system [1] and hyperspectral panoramic image stitching [2]. Many algorithms have been proposed for this issue [3,4]. Especially, due to the limitations of the imaging width, it is common that the ROI region cannot be contained in one view of a remote sensing image. Hence, it is necessary to capture multi-view and time-synchronized images from videos, then splice them into a panoramic image.

Accurate video synchronization aims at aligning different videos that capture the same event and share a temporal and visual overlap from multiple views; it is the de-facto industry standard for many scientific fields such as remote sensing image mosaic [3,4], human motion capture [5,6], optical flow estimation [7,8], group retrieval [9,10], dense 3D reconstructions [11,12], and spatial-temporal trajectory prediction modeling [13,14]. While recent advances allow moving from marker-based solutions to purely visual reconstruction [15], which alleviates actor instrumentation, multiple cameras are still required

for millimeter-level reconstruction. For instance, monocular human pose estimation has drastically improved in recent years [16–18]. However, the attained error is still above 3 cm on average, with occasional large outliers. A key factor is the unavoidable ambiguities in reconstructing a 3D scene from a 2D image with the depth information largely obscured. Therefore, the most accurate deep learning approaches that are used in neuroscience [19], sports, medical surgery [20], and other life science studies [21] rely on multiple views to reduce ambiguities. These multi-view solutions involve a calibration and a synchronization step before, or integrated into, the reconstruction algorithm.

Video synchronization can be solved in hardware, by wiring cameras together and by wireless solutions such as GPS, Bluetooth, and WiFi. However, the former is cumbersome to set up and unpractical for mobile equipment, and the latter requires special, expensive cameras. When recording with consumer cameras, external synchronization signals are common, such as a light flash or clap recorded on the audio line [22]. However, these are error prone and require manual interventions. Therefore, most practical synchronization pipelines require the user to click the occurrence of common events in every camera and video that should be synchronized, which is a time-intensive post-processing step for recordings with many sessions and cameras. By contrast, this paper aims at an automated yet general algorithm that matches the performance of existing domain-specific approaches.

We propose a new approach for learning the synchronization of multi-view videos that (1) is accurate, by using a new network architecture with motion trajectories as intermediate representations; (2) adapts to a diverse set of domains, as the required trajectory tracking is learned without supervision; and (3) is convenient to use, because no external calibration signals are needed.

Existing synchronization algorithms have experimented with a diverse set of video representations, ranging from raw RGB frames [23,24] over optical flow [25,26], and detecting the position of humans using off-the-shelf networks [27–30]. The former two are general but strike a lower accuracy. The latter works well for recording human motion, whereas they do not translate well to other instances when no pre-trained detectors are available. Our solution attempts to combine the best of both, by self-supervised learning of a sparse representation of the pictured scene into the location trajectories and appearance of moving salient objects and persons. The location tracks are integrated into a new neural network architecture that works on inferred sparse localization. Its advantage is that the motion that is important for synchronization can be disentangled from the appearance, which may vary across views and time due to illumination and viewing angle. The only supervision is the time annotation of a few example videos in the target domain, as in the prior work [26].

Our method is different from the method of Lorenz et al. [31], which uses a part-based disentangling method to generate new views by transferring the appearance in a specific view. This paper proposes a landmark-generating method named style transfer module by using the correspondence between two existing views, which provides useful positional information at different times. The style transfer module in the proposed method plays an important role in extracting features by temporal modeling, which fits the usage of the subsequent temporal similarity calculating module. To summarize, our main contribution lies in three folds:

- We propose a self-supervised style transfer solution that decomposes a scene into objects and their parts to learn domain-specific object position per frame that allows to track keypoint locations, such as animal position and articulated human pose over time.
- We propose an efficient two-stage method of style transfer and matrix diagonal (STMD) which uses the keypoint locations to train a generalized similarity model that can predict the synchronized offset between two views.
- Experimentations on three different video-synchronization datasets and the application of the image mosaic of UAV remote sensing prove the superiority and generalization of the proposed method on different domains.

2. Related Work

2.1. Synchronization Algorithms

Previous video synchronized methods use a diverse set of low-level and high-level motion features to infer a correlation between videos. Wu et al. [26] compare 2D human pose features with optical flow for training their Synchronization Network (SynNet) and find that the pose feature works better. Xu et al. [32] use the 3D pose as input and match the consistency of two-view pixel correspondences across video sequences. However, this requires a precise 3D reconstruction method. In addition, some works combine visual and auditive elements to realize video synchronization [22,33]. However, additional information such as audio sources may not be always available in real videos or be disrupted by diffuse background noise.

Wang et al. [34] propose a nonlinear temporal synchronization method using graph-based search algorithm with coefficient matrices to minimize the misalignments between two moving cameras. Different from their work, the proposed method is easier to conduct since it is self-supervised and does not need pre-trained information to obtain the correspondence between videos, while [34] needs to use predefined basis trajectories to obtain the coefficient matrices. Recently, Huo et al. [35] propose a reference frame alignment method for frame extrapolation to establish nonlinear temporal correspondence between videos. The proposed method is different from [35] since it is not dependent on supervised tracking and not sensitive to the error brought by tracking noise. Therefore, our method can adapt to various domains.

Another branch of related work finds implicit temporal correspondence without explicit motion features. Purushwalkam et al. [36] propose an alignment procedure to connect patches between videos via cross-video cycle consistency. Similarly, Dwibedi et al. [37] also apply temporal cycle consistency to align videos, but they use it to learn an embedding space to obtain the nearest neighbors. Other methods use some prior temporal mapping information (e.g., an event appeared in multiple videos) to learn some correspondence between multiple video sequences, such as ranking [38], Canonical Correlation Analysis [39], and co-occurring events [38,40]. However, these methods are not fit for our domain-adaptive task as this prior mapping information cannot exist in different scenarios.

2.2. Object Detection and Tracking

Traditional object detection methods need some manual object position annotations for supervised training [41–44] or body part annotation, such as OpenPose [30], which is widespread for humans but difficult for most other animals. For the tracking of people, Tompson et al. [45] propose a position refinement model to estimate the joint offset location and improve human localization. Newell et al. [46] propose associative embedding tags to track each keypoint for individual people. Recently, Ning et al. [47] use a skeleton-based representation of human joints to incorporate single-person pose tracking (SPT) and visual object tracking (VOT) as a unified framework. In addition, there are some works [48–50] that realize tracking in non-human cases, such as animals, which inspires us to generalize our method to the non-human cases of video synchronization, but does not yield the fine-grained resolution up to body parts that we desire.

2.3. Self-Supervised Methods

To tackle the problem without supervision, self-supervised learning (SSL) has been proposed to train the model using auxiliary tasks [51]. For object detection, SSL has been used to replace the ImageNet pretraining [52] by the relevant task that does not need manual annotation data, such as colorization [53], Jigsaw puzzles [54], inpainting [55], tracking [56], optical flow [57], temporal clues [58], text [59], and sound [60]. However, the majority of their performances are not as good as the pretraining of ImageNet. In addition, there are some works that use SSL in object detection by improving the auto-encoder network with the attention mechanism [61,62] or proposal-based segmentation [63]; these

approaches first use a spatial transform to detect bounding boxes and then pass them through the auto-encoder and synthesize the object with a background.

Different from the discussed previous work, we do not use any spatial supervision in this paper, yet derive high-level features that are better suited for synchronization than lower-level ones such as optical flow.

3. The Proposed Method

Generally speaking, the procedure of remote sensing panorama is summarized as five aspects: image registration, extraction of overlapping areas, radiometric normalization, seamline detection, and image blending. There are many similar aspects between panorama and video synchronization, e.g., finding internal correspondences among different overlapping views. According to the traditional procedure for remote sensing panorama, we propose a new video synchronization method as follows.

The proposed method operates in two steps as shown in Figure 1. The first stage estimates and tracks the coordinates of salient objects via a self-supervised network that is trained on the raw multi-view videos to establish correspondences. The second stage is a neural network that takes the object trajectories inferred from two videos as input, computes a similarity matrix across the two views, and predicts an offset based on these using classification into discrete classes.

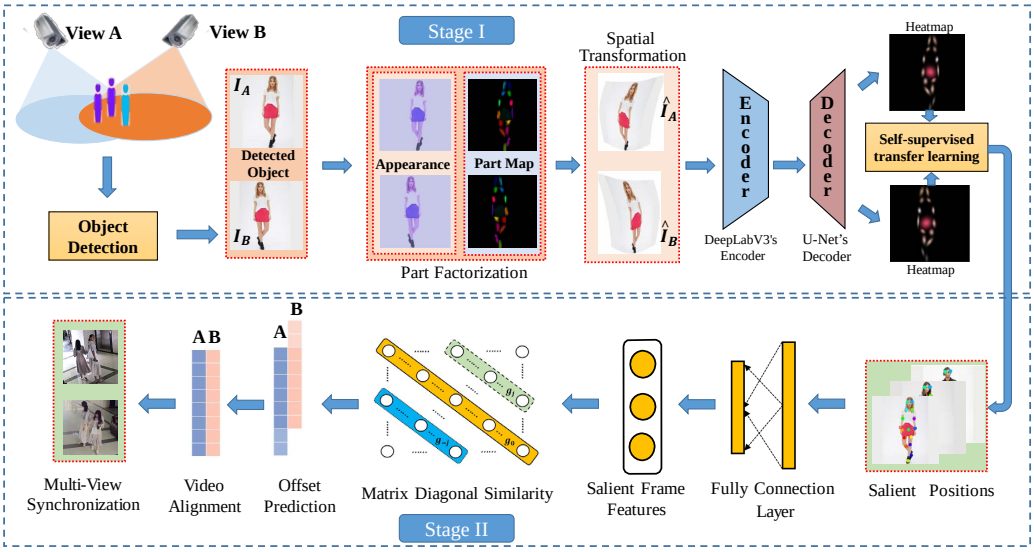

Figure 1. The overall pipeline of our proposed STMD video synchronization method. In the first step, raw images are processed with a self-supervised module that yields explicit object position and their trajectories over time. It is followed by a network tailored for the synchronization of the tracks from the first style transfer module. At the core of the synchronization network is a matrix diagonal module that measures the similarity over pairwise frames that correspond to the same temporal offset. The network is trained end-to-end on a classification objective.

3.1. Stage I: Style Transfer-Based Object Discovery and Tracking

To obtain the position of the salient objects in a video, we desire to divide each frame into an assembly of parts, defined by the 2D coordinate of the central point of each part. Many supervised approaches for the detection of objects and their parts are available. However, even though neural network architectures are sophisticated and attain high accuracy on the benchmarks, they poorly generalize to new domains. For instance, a method trained on persons will not generalize to animals, although positional and behavioral analysis is in high demand for application in neuroscience, medicine, and life sciences. Therefore, we tailor and extend the self-supervised approach from [31] to our domain before proceeding to the main goal of video synchronization.

The original idea of [31] is to disentangle pose and shape by training on pairs of images that share the same objects but have slight appearance variation and a different image constellation. A single image is turned into such a pair by adding color augmentation and spatial deformation via thin plate splines for the second example, which constructs the correspondence between two views by the style transfer of the image.

We consider the difference between two images taken from different viewpoints in a multi-view setup as a spatial image transformation $\tau : \Gamma \rightarrow \Gamma$, instead of relying on the explicit deformation that is difficult to parameterize. Therefore, we consider a pair of views as being composed of the same objects. Of course, the image transformation might have holes due to occlusions and the field of view of the two cameras will not overlap perfectly. Yet, we show that the following algorithm is robust to slight violations and works when these assumptions are approximately fulfilled.

Formally, we use a part-based factorization [31] to represent the object in an image I as Q parts:

$$\{\varphi_i(I)\} := (\varphi_1(I), \ldots \varphi_i(I), ..., \varphi_Q(I))^T \tag{1}$$

where the local part position is independent of other parts. Global image information is represented by the combinations of all individual parts $\{\varphi_i(I)\}$. Each part consists of a 2D position $\mathbf{u}_i \in \mathbb{R}^2$, shape $\mathbf{\Sigma}_i \in \mathbb{R}^2$, and its appearance encoding $\mathbf{a_i} \in \mathbb{R}^3$.

Part shape and appearance are learned by unsupervised learning as in [31], but we use multiple views instead of deformed variants of the same image. Let I_1 be an image from Camera 1 (Cam1) and I_2 be the image at the same time step in Camera 2 (Cam2), which is viewed as the geometry-transformed image of I_1. It is worth noting that since no explicit pose correspondence is used, the proposed ST module can also be trained with misaligned frames showing the same object in a slightly different pose (e.g., [31] trains with deformed images). Therefore, it is not necessary for the landmark generation to use extra annotations to make the images under two views aligned in the ST stage. Since the proposed subsequent MD module requires synchronized videos (cf. Section 1), we use the same synchronized footage for the ST module for simplicity here.

Color augmentation is used to create an appearance-transformed version of the two, $\hat{I}_1$ and $\hat{I}_2$. Thereby, I_1 can be reconstructed from the position in $\hat{I}_1$ and the color in I_2. The same holds for the other direction and we select one of the two at random. This reconstruction is realized with an autoencoder consisting of the DeepLabV3's encoder [64] and the U-Net's [65] decoder. Specifically, there are four up-sampling layers in the U-Net decoder, each layer consists of one deconvolution layer for upsampling and two ReLU convolution layers. The encoder is independently applied to each of the two images (($\hat{I}_1$, I_2) or ($\hat{I}_2$, I_1)) to realize semantic segmentation. The output feature maps are considered as a stack of heatmaps, one heatmap, $\mathbf{H}_i \in \mathbb{R}^{W \times H}$, where W and H are the width and height of the i'th part's heatmap. These heatmaps are normalized to form probability maps:

$$P_i(x, y) = \frac{\exp[\mathbf{H}_i(x, y)]}{\sum_{u=1}^{W} \sum_{v=1}^{H} \exp[\mathbf{H}_i(u, v)]} \tag{2}$$

where (u, v) and (x, y) are pixel locations. The position μ_i of part i is then computed as the expected 2D position, i.e., the weighted sum of all pixel locations, weighted by the probability map P_i. The shape, $\mathbf{\Sigma}_i$ is estimated as the covariance of P_i around μ_i. The appearance is estimated by creating a Gaussian map, $\mathbf{G}_i \in \mathbb{R}^{W \times H}$, with mean μ_i and covariance $\mathbf{\Sigma}_i$ and building the expected color over this distribution, i.e., the mean color value, weighted by the Gaussian support.

To decode the entire image, appearance and pose estimated from $\hat{I}_1$ and I_2 are mixed and converted into a color image by multiplying $\mathbf{a_i}$ with $\mathbf{G_i}$ and taking the maximum over all parts. This coarse image is blurry and is up-sampled to a proper image using U-Net as

a form of the decoder. This chain of the network is trained on a standard reconstruction objective comprised of a photometric pixel loss and a perceptual loss using VGG:

$$L_{rec} = \|I - I_{rec}\|_2 + \beta L_{perc}(I, I_{rec}) \tag{3}$$

where I_{rec} is the reconstructed image of I and β is the weight of perceptual loss.

In total, the first stage uses self-supervision to learn domain-specific object position per frame that allows tracking keypoint locations, such as animal position and articulated human pose over time. To this end, we rely on existing self-supervised solutions that decompose a scene into objects and their parts by finding an association between a training image and its appearance and spatially deformed twin. We utilize a similar training framework but learn the disentanglement on a pair of images from different videos picturing the same scene instead of a deformed version of the same image. This establishes correspondences across views and circumvents the use of deformation models that are difficult to tune.

3.2. Stage II: Matrix Diagonal Similarity-Based Classification Framework

After obtaining the positions of salient points, we propose to feed them into a matrix diagonal (MD) module that scores the alignment of videos.

The goal of video synchronization is to achieve temporal offset between two unaligned videos, where the video consists of many discrete images with a fixed frame rate. Therefore, the video synchronization problem is framed as a classification problem with quantified integer offset values: $\{-K, -K+1, \ldots, -1, 0, 1, \ldots, K\}$, where K is the half clip length and $K > 0$, there are $2K+1$ class labels to formulate the possible offsets. In this way, if we find the offset between two video frames, these can be aligned by shifting with the predicted offset.

Let $\mathbf{k}_{c_1,i} \in \mathbb{R}^D$ and $\mathbf{k}_{c_2,j} \in \mathbb{R}^D$ be features of the ith frame and jth frame from Cam1 and Cam2, respectively. In our full model, the features are the positions of the parts learned in the previous section, but we also compare with other features used in related work. Each raw feature is further processed with a matching network f to the refined features $\mathbf{e}_{c_1,i} \in \mathbb{R}^{D'}$ and $\mathbf{e}_{c_2,j} \in \mathbb{R}^{D'}$. D and D' are the respective spatial dimension. The network f consists of two FC-layers of width $[N_1, N_2]$. To compute a similarity between these features, we arrange them in a matrix of all possible feature pairs and compute their pairwise similarity,

$$M_{m,n} = -\frac{1}{l}||\mathbf{e}_{c_1,i+m} - \mathbf{e}_{c_2,j+n}||_2^2 \tag{4}$$

where the mean square error (MSE) is used to represent the feature distance between two frames, and the negative MSE value is used to measure the similarity between them. As shown in Figure 2, l is the length of the clip. We set the clip $C_1 = \{\mathbf{e}_{c1,i}, \ldots, \mathbf{e}_{c1,i+2K-1}\}$ and the clip $C_2 = \{\mathbf{e}_{c2,j}, \ldots, \mathbf{e}_{c2,j+2K-1}\}$ as the element of the row and the column in the matrix $\mathbf{M}$, respectively. In this way, we compute the similarity of all frames between two clips C_1 and C_2 to obtain Matrix $\mathbf{M}$.

With this similarity matrix computed, we find the offset with the highest similarity. Since all frames are recorded with the same frame rate, a temporal shift of t corresponds to matching frames in the g_t'th off-diagonal of $\mathbf{M}$. In the case of two synchronized clips, the minimum should appear in the main diagonal. Thus, the average similarity along diagonals of M is computed as

$$S_{C_1,C_2} = \frac{1}{l_t} \sum_{(m,n)\in g_t} M_{m,n} \tag{5}$$

where l_t is the length of diagonal g_t. Finally, we compute the offset T between two input video clips according to the distance between the main diagonal and the diagonal that has the maximum average similarity. In this way, the two input videos can be synchronized by shifting the offset.

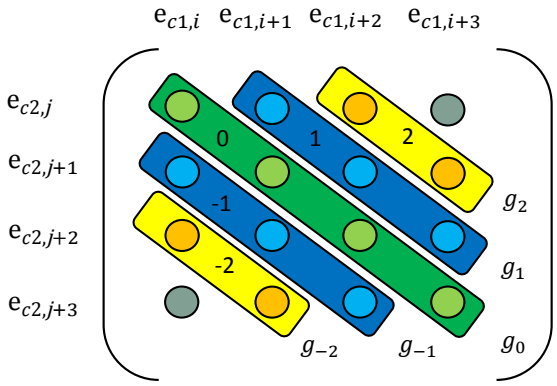

Figure 2. An illustration of the matrix diagonal similarity-based classification framework. The matrix size is 4×4 and $K = 2$, the clip length is 4, diagonals with different colors represent the corresponding offset, and each circle represents the matrix element $M_{m,n}$.

The feature extraction network f is trained end-to-end on a cross-entropy loss, given ground truth offset labels, we use two fully connection layers to encode the 2D coordinates of salient positions into frame features, and there is a ReLU layer between the two layers. In addition, the detected landmarks are ordered and generally consistent between two views in the output of the encoder in the style transfer (ST) stage, e.g., the same keypoint is always on the human head. Therefore, even if a given part is not identified in one of the images in some extreme cases, the MD stage includes a learned neural network and can hence rely on this ordering to avoid the remaining features being shifted, then ensures its robustness.

4. Experiments

In this section, we demonstrate the accuracy and generality of the proposed approach to video synchronization datasets. Besides the simulated Cube&Sphere dataset, we conduct experiments on another two datasets: One dataset is collected from two views of the Human 3.6 Million (Human3.6M) dataset [66,67], an established benchmark for 3D human pose estimation with synchronized videos. The other is a custom dataset that resembles capture setups of neuroscience laboratory animals. We refer to our full method as STMD in experiments and compare against diverse baselines. To make the experiments fair and convincing, we used the cross-validation method to evaluate and obtain average results. Specifically, to evaluate its generalization, the proposed video synchronization method will be conducted in some practical remote sensing fields, e.g., the UAV image mosaic.

4.1. Datasets

Cube&Sphere Video Synchronization Dataset. The Cube&Sphere dataset is constructed using the open-source 3D animation suite Blender. We generated 60 random 3D positions of a cube and a sphere. This scene is captured from two cameras with a view angle difference of roughly 30 degrees. Each video is 1200 frames long, with a frame rate of 24 fps. The first 960 frame pairs were used to construct the training dataset, and the last 240 were used for testing. The 3D coordinates of the virtual objects were projected onto the 2D image plane of the two cameras to form positions for a supervised baseline.

Fish Video Synchronization Dataset. We chose a pair of synchronized clips from 2 views and each consisted of 256 successive frames at 30 fps from a neuroscience experiment setup with a zebrafish (*Danio rerio*) in random motion, as shown in Figure 3. The first 128 frames were used for training, while the last 128 frames were used for testing. To ignore motions in the background, only the fish tank region was used as input to the algorithm.

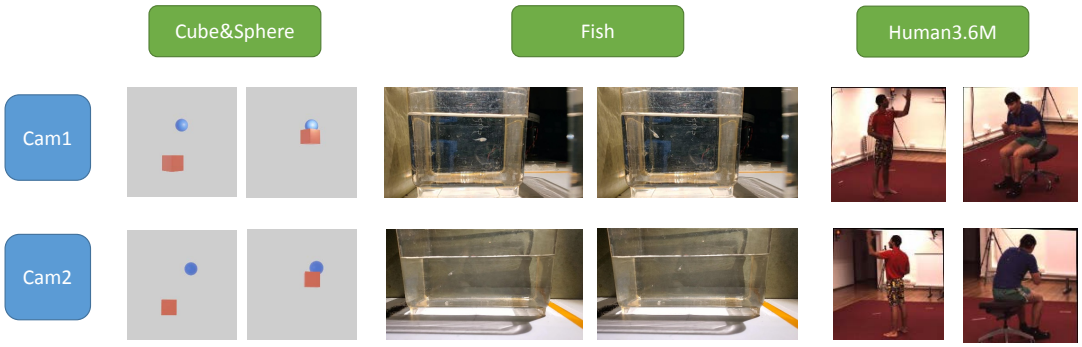

Figure 3. Illustration of the three video synchronization dataset. Cam1 and Cam2 in Rows 1 and 2 are the corresponding two views aligned at the same time point.

Human3.6M Video Synchronization Dataset. We use the well-known Human3.6M dataset, which contains recordings of 11 subjects with four fully-synchronized and high-resolution progressive scan cameras at 50 Hz [66]. We use 60 sequences from two cameras of Human3.6M, which includes walking, sitting, waiting, and lying down. There are 720 frames in each camera, we use the first 540 for training and the last 180 for testing, the size of each image is 128 × 128.

4.2. Metrics

To provide a fair comparison with other methods, we use the well-known Cumulated Matching Characteristic (CMC) [26,68] to report the synchronized accuracy results. It measures the top dist-k (dk) accuracy of k-different synchronized offsets. Moreover, we complement another SynError metric to measure their time deviation between the predicted offset R_i and the true offset T_i at the i-th frame as [26]:

$$SynError = \left(\frac{1}{L} \sum_{i=1}^{N} |R_i - T_i| \right) \times \frac{ds}{fps} \tag{6}$$

where L is the length of the video clip, ds is the video downsampling rate, and fps is the frame rate of the video.

4.3. Experiment Setup

Our proposed STMD is implemented using Pytorch. In the style transfer stage, we use the DeepLabV3 model [64] with a ResNet-50 backbone [69] to segment the Gaussian parts from the original images, and set the learning rate at 10^{-3}, the numbers of salient points are 13, 3, and 15 for Cube&Sphere, Fish, and Human3.6M dataset, respectively.

In the matrix diagonal stage, we use the cross-entropy as the loss criterion and the Adam method [70] for stochastic optimization over 50 epochs over the training set with a learning rate at 10^{-4}, and the neurons of the two FC-layers $[N_1, N_2] = [240, 168]$.

4.4. Results on Cube&Sphere Dataset

To provide a wide perspective of the performance of our proposed method, we present our results along with some start-of-the-art baselines and ablation studies on the Cube&Sphere dataset in Table 1. We reproduce the SynNet method by both the OpenPose strategy as [26] and the ST strategy, the former trains OpenPose from scratch to get keypoints according to the structure of human motion, in this case, Openpose extracts the heatmap of keypoints by human pose estimation. For the sake of illustration, we name it SynNet+OpenPose. Meanwhile, the latter uses our proposed ST module to obtain the keypoints and feed them into SynNet. In the experiment, both of them play the role of transfer, SynNet+OpenPose transfers the pre-trained human joints model on the keypoint estimation of the non-human case, while SynNet+ST uses the proposed style transfer between two camera views to generate non-human keypoints. Furthermore, we conduct

the ablation study in terms of the ST and MD modules, respectively. GTpoint+MD uses the geometric central points to substitute the ST module, which are set by Blender software to handle the motion of the objects. While SynNet+ST uses SynNet after the ST module to predict the offset rather than the MD module.

Table 1. Results of different methods on the Cube&Sphere dataset. "ds" represents the downsampling rate, the baselines without the "ds" label are ds = 1 by default.

Method	test-d0(%)	test-d1(%)	test-d2(%)	test-d3(%)	SynError ↓
SynNet+OpenPose [26]	10.9	31.2	50.2	68.8	0.1389
SynNet+ST	21.8	36.8	54.6	67.9	0.0782
LAMV(ds = 1) [71]	30.0	50.1	70.2	90.2	0.0638
PE [72]	34.3	67.7	72.1	79.6	0.0672
LAMV(ds = 2)	41.7	58.5	75.2	92.0	0.1061
LAMV(ds = 3)	50.1	64.5	78.9	93.1	0.1362
LAMV(ds = 4)	56.4	69.0	81.5	94.1	0.1586
TCC(ds = 1) [37]	67.1	81.8	89.6	92.9	0.0305
GTpoint+MD	77.1	99.0	99.4	99.5	0.0098
TCC(ds = 2)	80.9	88.1	91.4	92.8	0.0505
STMD+MSE	86.1	99.0	99.6	99.7	0.0059

From the results, we can draw the following conclusions that validate the improvements gained from our contributions.

- The SynNet method [26] uses OpenPose [30], which outputs a heatmap for each human body joint. It is similar to our proposed method that disentangles the image into parts, but does not generalize to general objects since the detector is trained on humans. By contrast, our ST module precisely estimates the salient points of the non-human object that are shown in Figure 4, which showcases the better generalization of our self-supervised approach.
- To facilitate a fair comparison of the SYN network architecture and our synchronization network, we use heatmaps generated by ST as input to train SynNet. We call this combination with our self-supervised part maps (SynNet+ST). It improves the accuracy of SynNet+OpenPose by 10.9%. Moreover, our full method attains a higher offset prediction accuracy, which shows that operating on explicit 2D positions and their trajectories is better than using discretized heatmaps as input (as used in SynNet+ST).
- In addition, we also compare against the GTpoint+MD and PE methods. The former is a strong baseline that uses the ground truth 2D coordinates instead of estimated ones to compute the matrix diagonal similarity. These GT positions are the central positions of the cube and sphere in Blender, projected onto the image plane. The latter uses positional encoding (PE) [72] on the ground truth 2D positions. These are projections of the 2D point onto sinusoidal waves of different frequencies, providing a smooth and hierarchical encoding of positions. We try using the PE strategy before the coordinate feature is fed to MD to make a comparison with our absolutely coordinate feature in STMD, the proposed STMD+MSE surpasses them by a large margin, which infers that using the original absolute position generated by the ST stage is better than Blender and PE in MD stage.
- Finally, we also compare with some baselines using different downsampling rates. The results in Table 1 show that the testing accuracy is increased while the SynError is decreased, which infers that the downsampling strategy improves accuracy by sacrificing SynError. Moreover, the proposed STMD method outperforms all the downsampling cases of other baselines, and the proposed ST module can improve the MD module with the GT coordinates from Blender (GTpoint+MD) with 9.0% test-d0, which validates the superiority of our method.

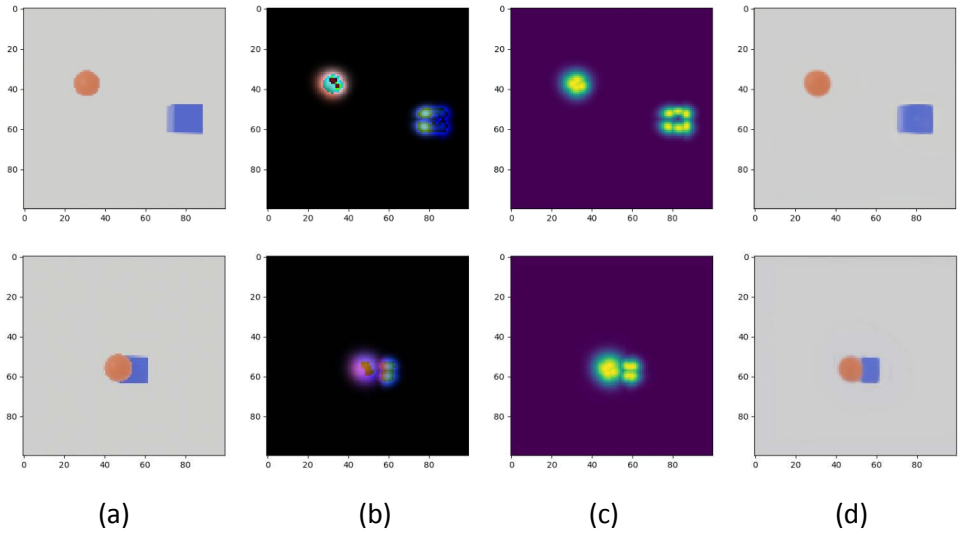

Figure 4. Some illustrations of ST module on Cube. (**a**) Original frame. (**b**) Part-based model. (**c**) Part-based heatmap. (**d**) Reconstruction frame.

4.5. Results on Fish Dataset

The Fish dataset is challenging, as the fish is small compared to the entire image, has a similar color to the fish tank glass wall, and has a smooth rather than crisp appearance. These factors pose difficulties for the style transfer module. To alleviate the impact of the changing background, we compute the background as the median pixel value over 100 frames spaced over each video. This background is subtracted from each frame. Note that the low color contrast leads to remaining artifacts. However, as the following analysis shows, the entire pipeline is robust to slight inaccuracies in object detection. We refer to background subtracted variants with the addition (Sub) to the network name.

The visualization of the results obtained by the ST model is shown in Figure 5. Without background subtractions, the localization fails (Row 1). After extracting the foreground and feeding these cleaned images to the ST stage, we obtain more precise salient points that track the fish well (Row 2). As shown in Table 2, the test-d0 reaches 94.5% when ST epoch = 190. Moreover, as shown in Figure 6a, we also plot the tendency curve to illustrate the performance of MD in 190 ST epochs. To better display the result, we use the moving average result in Figure 6a, the moving window size is 25. All the curves gain a large margin as the epoch increases, which validates the robustness of our method.

Table 2. Results of STMD with subtraction on the Fish dataset with different epochs of ST. The number in the bracket after STMD is the training epochs of the ST stage.

Method	test-d0(%)	test-d1(%)	test-d2(%)	test-d3(%)	SynError ↓
TCC(ds = 1) [37]	16.4	25.2	34.1	45.1	0.1042
STMD(1)	18.4	33.9	38.5	47.7	0.1581
LAMV(ds = 1) [71]	21.0	35.1	49.5	64.1	0.0864
TCC(ds = 2)	28.3	36.7	47.4	58.9	0.1646
LAMV(ds = 2)	30.7	43.2	56.1	68.5	0.1512
STMD(50)	34.9	64.2	64.2	70.6	0.1248
TCC(ds = 3)	35.0	47.8	60.2	73.0	0.1785
LAMV(ds = 3)	38.4	49.8	60.8	71.9	0.2013
LAMV(ds = 4)	44.8	54.7	64.7	74.8	0.2410
TCC(ds = 4)	48.7	58.1	69.4	79.7	0.1766
STMD(100)	78.9	78.9	88.1	91.7	0.0229
STMD(150)	88.1	92.4	92.6	92.7	0.0119
STMD(190)	94.5	98.1	98.3	99.1	0.0080

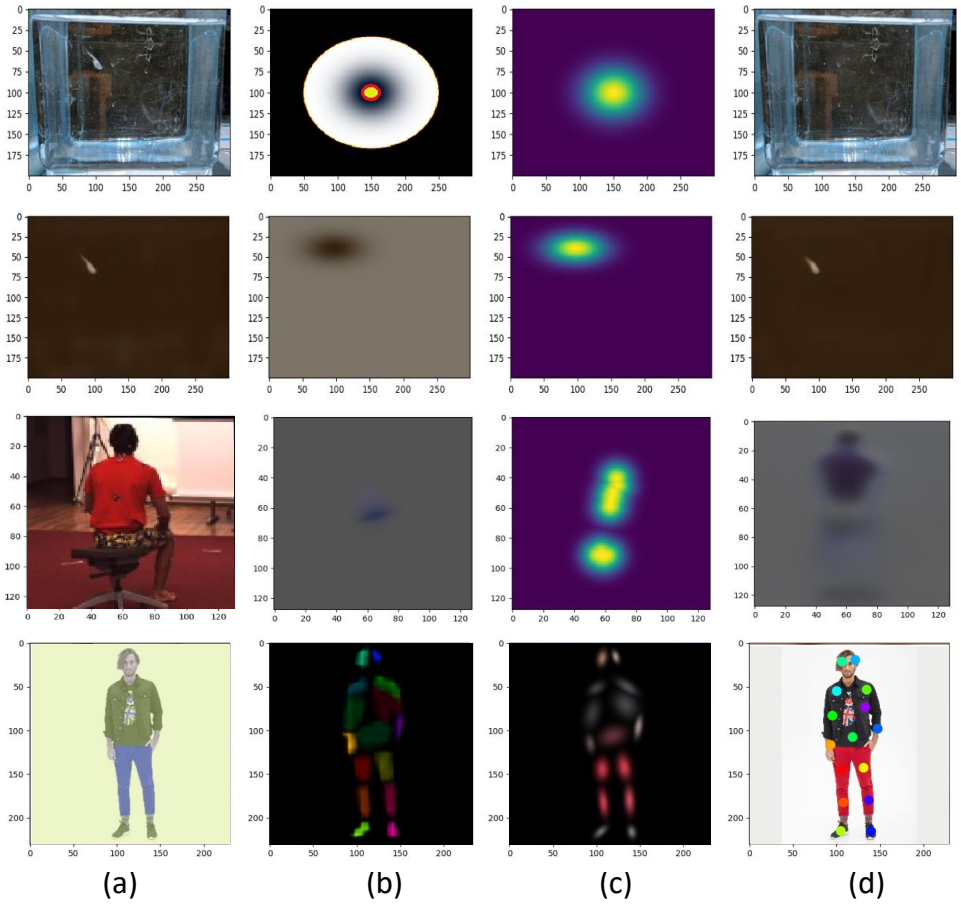

Figure 5. Some illustrations of the ST module on fish (Row 1–2) and human scenarios (Row 3–4). As a reference, we show the results of the source frame and subtraction for comparison. (**a**) The original frame (Row 1 and 3) or background subtraction (Row 2 and 4). (**b**) Part-based model. (**c**) Part-based heatmap. (**d**) Reconstruction frame.

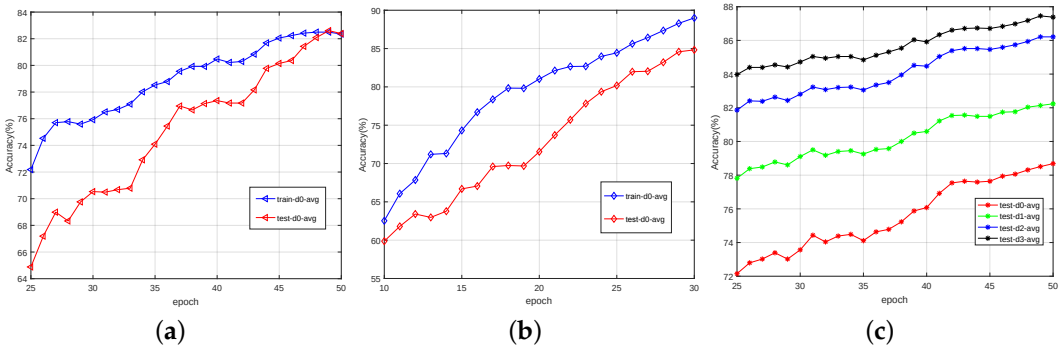

Figure 6. (**a**) The video synchronization results on the Fish dataset. (**b**) The video synchronization results of STMD (Sub) plotted over 30 ST epochs on Human3.6M dataset. (**c**) The video synchronization results w/o Sub on Human3.6M dataset.

4.6. Results on Human36M Dataset

The video synchronization results are shown in Table 3. The testing accuracy was computed for the best-performing snapshot computed over 50 training epochs. The best accuracy of test-d0 reached 88.2%. The proposed method scored highest, with the same order as observed for the simpler Cube&Sphere dataset.

We compared the performance of the original image and the post-procession of subtraction (Sub) for the STMD method. There was a leap in improvement from the subtraction

to the proposed method, by 9.9% for the test-d0 accuracy. This result validates the visual improvements shown in Figure 5 (third vs. fourth row), leading to the ST module focusing more on the moving object rather than rich textures in the background.

Table 3. Results on the Human36M dataset; 0–3 represent the testing accuracy of the respective predicted offset, SynNet uses OpenPose [30] to extract the pose feature as [26], and "PE" denotes the variant using positional encoding to represent 2D object positions [72].

Method	test-d0(%)	test-d1(%)	test-d2(%)	test-d3(%)	SynError ↓
SynNet [26]	14.5	23.5	38.0	47.5	0.3038
PE [72]	20.9	40.5	55.7	66.0	0.2055
TCC(ds = 1) [37]	28.4	40.2	53.7	63.4	0.3154
PE(sub)	31.1	45.3	59.0	73.3	0.2000
TCC(ds = 2)	38.3	50.0	59.6	71.0	0.3400
TCC(ds = 3)	45.6	54.4	65.3	72.5	0.3801
TCC(ds = 4)	51.0	61.4	68.0	75.6	1.6402
LAMV(ds = 1) [71]	54.7	60.7	66.2	69.9	0.2226
LAMV(ds = 4)	62.9	64.9	66.6	69.7	1.1553
LAMV(ds = 2)	66.8	70.9	74.3	77.2	0.3763
LAMV(ds = 3)	67.9	70.0	72.0	73.7	0.7385
STMD	78.3	82.6	85.7	86.3	0.0963
STMD(sub)	88.2	93.1	93.3	94.4	0.0485

In addition, we conduct experiments to monitor the training curves of the proposed model. We plot the synchronized performance with different epochs for both the ST and MD modules.

Moreover, to test the effectiveness of the ST module and observe the trend in more detail, we plotted the first 30 epochs with a moving average of window size 10. As shown in Figure 6b, both the training and testing accuracy curves keep increasing with more training epochs, which validates the effectiveness of the ST training model.

We also evaluate the training curve over 50 epochs of the MD module in Figure 6c with a moving average window size of 25. To validate the robustness of the proposed STMD method, we use the ST model without subtraction to evaluate. The test-d0 accuracy is the most important indicator in video synchronization, yet the others are auxiliary to analyze consistency. Figure 6c plots the corresponding testing results. All metrics kept increasing with the number of epochs, which validates the robust transfer ability from training to testing.

4.7. Limitations

Figure 7 shows some failure cases observed during our experiments on the three datasets. Given two unsynchronized input clips, we predict the offsets and adjust them to synchronize. From Figure 7 and others inspected, the wrong predictions mainly occurred in the hard case of large offsets or existing severe occlusions, e.g., Figure 7a. It violates the assumption that a pair of views should be composed of the same objects in Section 3.1, which is hard to predict the precise frame offset because salient features are missing. We observe that our method still predicts the correct direction of offset in all the above hard cases, which validates that the proposed STMD method can still work within a certain margin of synchronization error.

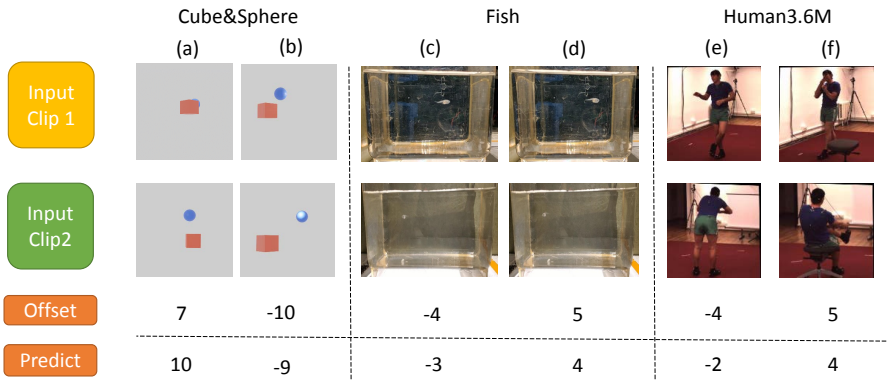

Figure 7. Illustration of representative failure cases. Images in Rows 1–2 represent the first frame of the video clips from different views, respectively. Row 3 annotates the ground truth offset between clip2 and clip1, negative value denotes clip2 lags behind clip1, and Row 4 gives our predicted offset. The offset range is [−10,10] for the Cube&Sphere dataset and [−5,5] for the other 2 datasets. (**a**–**f**) represent six pairs of clips to display their ground truth offsets and our predicted offsets.

4.8. STMD Method for the UAV Remote Sensing Image Mosaic

To validate the practical performance of the proposed video synchronization method, we apply it on the remote sensing applications of reconstructing the panorama of the aerial image taken by an Unmanned Aerial Vehicle (UAV). Such UAV remote sensing image mosaic technique plays important roles in many fields such as forestry, agriculture, and soil resources. In this setting, the proposed video synchronization method can provide useful matching information of salient points among multiple views by self-supervised scene decomposition, as shown in Figure 8. The left three image sequences were collected by an UAV with minor time offsets; therefore, there were many overlapping areas among the images, which is closely related to multi-view video synchronization under the moving cameras. Hence, the salient positions between the pairwise perspective can be captured and matched by the style transfer module in our video synchronization method. Based on these common features, the images can be spliced together to a wider view. In this way, the proposed video synchronization can be used in image mosaic with a certain time range of slight offsets to obtain the panoramic aerial image, which is illustrated in Figure 8d, by self-supervised learning the correspondence among salient points effectively.

Figure 8. The illustration of image mosaic for UAV remote sensing panorama by the proposed video synchronization method. Some detected salient positions among views are displayed and matched by lines. (**a**–**c**) display three image sequences with minor time offsets, (**d**) shows the image stitching result.

5. Conclusions

This paper presents STMD, an efficient two-stage video synchronization method that can easily be adapted to new domains by learning domain-adaptive motion features from multiple views without requiring any spatial annotation. The gains in synchronization accuracy are due to the joint contribution of this self-supervised pre-processing, and a matrix diagonal module-based network architecture is tailored to predict the temporal offset from 2D trajectories. Our experiments show the superiority of our method. It can be generalized to practical settings such as remote sensing application. It is worth mentioning that this paper treats video synchronization as a classification problem, it selects on the frame level and does not include the sub-frame level synchronization.

In future, there are three directions that can be conducted to expand the work. At first, more complicated fields such as the fish swarm scenario can be considered in the synchronization task. Furthermore, this paper mainly proposes a 2D video synchronization work, we will try to use the 3D trajectory to model the perspective and handle the occlusion problem. Finally, more precise methods can be proposed to take the synchronization of the sub-frame level into account, which makes the work more practical to the real application.

Author Contributions: Conceptualization, L.M. and H.R.; methodology, L.M., Y.H., F.J.F. and H.R.; software, L.M. and Y.H.; validation, L.M.; formal analysis, L.M.; investigation, L.M. and Y.Y.; resources, H.R., L.Z. and Y.Y.; data curation, L.M. and Y.H.; writing—original draft preparation, L.M.; writing—review and editing, H.R. and Y.Y.; visualization, L.M., Y.H. and F.J.F.; supervision, H.R. and Y.Y.; project administration, H.R., L.Z. and Y.Y.; funding acquisition, Y.Y. and L.Z. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by the Fundamental Research Funds for the Central Universities, HUST: 2020kfyXJJS045, and the International Program Fund for Young Talent Scientific Research People, Sun Yat-sen University .

Data Availability Statement: In this paper, the Cube&Sphere dataset, Fish dataset, and Human36M dataset are employed for experimental verification. Readers can obtain these datasets from the author by email (meil3@mail2.sysu.edu.cn).

Acknowledgments: The authors gratefully acknowledge Ying-cong Chen for his linguistic assistance during the preparation of this manuscript. He is an Assistant Professor of Hong Kong University of Science and Technology. He was a Postdoctoral Associate at Computer Science & Artificial Intelligence Lab of Massachusetts Institute of Technology (MIT), USA.

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

References

1. Aires, A.S.; Marques Junior, A.; Zanotta, D.C.; Spigolon, A.L.D.; Veronez, M.R.; Gonzaga, L., Jr. Digital Outcrop Model Generation from Hybrid UAV and Panoramic Imaging Systems. *Remote Sens.* **2022**, *14*, 3994. [CrossRef]
2. Zhang, Y.; Mei, X.; Ma, Y.; Jiang, X.; Peng, Z.; Huang, J. Hyperspectral Panoramic Image Stitching Using Robust Matching and Adaptive Bundle Adjustment. *Remote Sens.* **2022**, *14*, 4038. [CrossRef]
3. Han, P.; Ma, C.; Chen, J.; Chen, L.; Bu, S.; Xu, S.; Zhao, Y.; Zhang, C.; Hagino, T. Fast Tree Detection and Counting on UAVs for Sequential Aerial Images with Generating Orthophoto Mosaicing. *Remote Sens.* **2022**, *14*, 4113. [CrossRef]
4. Hwang, Y.S.; Schlüter, S.; Park, S.I.; Um, J.S. Comparative evaluation of mapping accuracy between UAV video versus photo mosaic for the scattered urban photovoltaic panel. *Remote Sens.* **2021**, *13*, 2745. [CrossRef]
5. Wandt, B.; Little, J.J.; Rhodin, H. ElePose: Unsupervised 3D Human Pose Estimation by Predicting Camera Elevation and Learning Normalizing Flows on 2D Poses. In Proceedings of the 2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition, New Orleans, LA, USA, 18–24 June 2022.
6. Gholami, M.; Wandt, B.; Rhodin, H.; Ward, R.; Wang, Z.J. AdaptPose: Cross-Dataset Adaptation for 3D Human Pose Estimation by Learnable Motion Generation. In Proceedings of the 2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition, New Orleans, LA, USA, 18–24 June 2022.
7. Mei, L.; Lai, J.; Xie, X.; Zhu, J.; Chen, J. Illumination-invariance optical flow estimation using weighted regularization transform. *IEEE Trans. Circuits Syst. Video Technol.* **2019**, *30*, 495–508. [CrossRef]
8. Mei, L.; Chen, Z.; Lai, J. Geodesic-based probability propagation for efficient optical flow. *Electron. Lett.* **2018**, *54*, 758–760. [CrossRef]

9. Mei, L.; Lai, J.; Feng, Z.; Xie, X. From pedestrian to group retrieval via siamese network and correlation. *Neurocomputing* **2020**, *412*, 447–460. [CrossRef]
10. Mei, L.; Lai, J.; Feng, Z.; Xie, X. Open-World Group Retrieval with Ambiguity Removal: A Benchmark. In Proceedings of the 25th International Conference on Pattern Recognition, Milan, Italy, 10 January–15 January 2021; pp. 584–591.
11. Mahmoud, N.; Collins, T.; Hostettler, A.; Soler, L.; Doignon, C.; Montiel, J.M.M. Live tracking and dense reconstruction for handheld monocular endoscopy. *IEEE Trans. Med. Imaging* **2018**, *38*, 79–89. [CrossRef]
12. Zhen, W.; Hu, Y.; Liu, J.; Scherer, S. A joint optimization approach of lidar-camera fusion for accurate dense 3-d reconstructions. *IEEE Robot. Autom. Lett.* **2019**, *4*, 3585–3592. [CrossRef]
13. Huang, Y.; Bi, H.; Li, Z.; Mao, T.; Wang, Z. Stgat: Modeling spatial-temporal interactions for human trajectory prediction. In Proceedings of the 2019 IEEE/CVF International Conference on Computer Vision, Seoul, Republic of Korea, 27 October–2 November 2019; pp. 6272–6281.
14. Sheng, Z.; Xu, Y.; Xue, S.; Li, D. Graph-based spatial-temporal convolutional network for vehicle trajectory prediction in autonomous driving. *IEEE Trans. Intell. Transp. Syst.* **2022**, *23*, 17654–17665. [CrossRef]
15. Saini, N.; Price, E.; Tallamraju, R.; Enficiaud, R.; Ludwig, R.; Martinovic, I.; Ahmad, A.; Black, M.J. Markerless outdoor human motion capture using multiple autonomous micro aerial vehicles. In Proceedings of the 2019 IEEE/CVF International Conference on Computer Vision, Seoul, Republic of Korea, 27 October–2 November 2019; pp. 823–832.
16. Rhodin, H.; Spörri, J.; Katircioglu, I.; Constantin, V.; Meyer, F.; Müller, E.; Salzmann, M.; Fua, P. Learning monocular 3D human pose estimation from multi-view images. In Proceedings of the 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 8437–8446.
17. Sharma, S.; Varigonda, P.T.; Bindal, P.; Sharma, A.; Jain, A. Monocular 3D human pose estimation by generation and ordinal ranking. In Proceedings of the 2019 IEEE/CVF International Conference on Computer Vision, Seoul, Republic of Korea, 27 October–2 November 2019; pp. 2325–2334.
18. Ci, H.; Ma, X.; Wang, C.; Wang, Y. Locally connected network for monocular 3D human pose estimation. *IEEE Trans. Pattern Anal. Mach. Intell.* **2020**, *44*, 1429–1442. [CrossRef]
19. Graa, O.; Rekik, I. Multi-view learning-based data proliferator for boosting classification using highly imbalanced classes. *J. Neurosci. Methods* **2019**, *327*, 108344. [CrossRef]
20. Ye, M.; Johns, E.; Handa, A.; Zhang, L.; Pratt, P.; Yang, G.Z. Self-Supervised Siamese Learning on Stereo Image Pairs for Depth Estimation in Robotic Surgery. *arXiv* **2017**, arXiv:1705.08260.
21. Zhuang, X.; Yang, Z.; Cordes, D. A technical review of canonical correlation analysis for neuroscience applications. *Hum. Brain Mapp.* **2020**, *41*, 3807–3833. [CrossRef]
22. Wang, J.; Fang, Z.; Zhao, H. AlignNet: A Unifying Approach to Audio-Visual Alignment. In Proceedings of the 2020 IEEE Winter Conference on Applications of Computer Vision, Snowmass, CO, USA, 1–5 March 2020; pp. 3309–3317.
23. Wang, O.; Schroers, C.; Zimmer, H.; Gross, M.; Sorkine-Hornung, A. Videosnapping: Interactive synchronization of multiple videos. *ACM Trans. Graph. (TOG)* **2014**, *33*, 1–10. [CrossRef]
24. Wieschollek, P.; Freeman, I.; Lensch, H.P. Learning robust video synchronization without annotations. In Proceedings of the 2017 16th IEEE International Conference on Machine Learning and Applications (ICMLA), Cancun, Mexico, 18–21 December 2017; IEEE: Piscataway, NJ, USA, 2017; pp. 92–100.
25. Wang, L.; Xiong, Y.; Wang, Z.; Qiao, Y.; Lin, D.; Tang, X.; Van Gool, L. Temporal segment networks: Towards good practices for deep action recognition. In Proceedings of the European Conference on Computer Vision, Amsterdam, The Netherlands, 8–16 October 2016; Springer: Cham, Switzerland, 2016; pp. 20–36.
26. Wu, X.; Wu, Z.; Zhang, Y.; Ju, L.; Wang, S. Multi-Video Temporal Synchronization by Matching Pose Features of Shared Moving Subjects. In Proceedings of the 2019 IEEE/CVF International Conference on Computer Vision Workshops, Seoul, Republic of Korea, 27 October–2 November 2019; pp. 2729–2738.
27. Du, Y.; Wang, W.; Wang, L. Hierarchical recurrent neural network for skeleton based action recognition. In Proceedings of the 2015 IEEE/CVF Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, 7–12 June 2015; pp. 1110–1118.
28. Shahroudy, A.; Liu, J.; Ng, T.T.; Wang, G. Ntu RGB+D: A large scale dataset for 3D human activity analysis. In Proceedings of the 2016 IEEE/CVF Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 1010–1019.
29. Cao, Z.; Simon, T.; Wei, S.E.; Sheikh, Y. Realtime multi-person 2D pose estimation using part affinity fields. In Proceedings of the 2017 IEEE/CVF Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 7291–7299.
30. Cao, Z.; Martinez, G.H.; Simon, T.; Wei, S.E.; Sheikh, Y.A. OpenPose: Realtime Multi-Person 2D Pose Estimation using Part Affinity Fields. *IEEE Trans. Pattern Anal. Mach. Intell.* **2021** *43*, 172–186. [CrossRef]
31. Lorenz, D.; Bereska, L.; Milbich, T.; Ommer, B. Unsupervised part-based disentangling of object shape and appearance. In Proceedings of the 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 15–20 June 2019; pp. 10955–10964.
32. Xu, X.; Dunn, E. Discrete Laplace Operator Estimation for Dynamic 3D Reconstruction. In Proceedings of the 2019 IEEE/CVF International Conference on Computer Vision, Seoul, Republic of Korea, 27 October–2 November 2019; pp. 1548–1557.
33. Korbar, B. Co-Training of Audio and Video Representations from Self-Supervised Temporal Synchronization. Master's Thesis, Dartmouth College: Hanover, NH, USA, 2018.

34. Wang, X.; Shi, J.; Park, H.S.; Wang, Q. Motion-based temporal alignment of independently moving cameras. *IEEE Trans. Circuits Syst. Video Technol.* **2016**, *27*, 2344–2354. [CrossRef]
35. Huo, S.; Liu, D.; Li, B.; Ma, S.; Wu, F.; Gao, W. Deep network-based frame extrapolation with reference frame alignment. *IEEE Trans. Circuits Syst. Video Technol.* **2020**, *31*, 1178–1192. [CrossRef]
36. Purushwalkam, S.; Ye, T.; Gupta, S.; Gupta, A. Aligning videos in space and time. In Proceedings of the European Conference on Computer Vision, Glasgow, UK, 23–28 August 2020; Springer: Cham, Switzerland, 2020; pp. 262–278.
37. Dwibedi, D.; Aytar, Y.; Tompson, J.; Sermanet, P.; Zisserman, A. Temporal cycle-consistency learning. In Proceedings of the 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 15–20 June 2019; pp. 1801–1810.
38. Sermanet, P.; Lynch, C.; Chebotar, Y.; Hsu, J.; Jang, E.; Schaal, S.; Levine, S.; Brain, G. Time-contrastive networks: Self-supervised learning from video. In Proceedings of the 2018 IEEE International Conference on Robotics and Automation (ICRA), Brisbane, Australia, 21–25 May 2018; pp. 1134–1141.
39. Andrew, G.; Arora, R.; Bilmes, J.; Livescu, K. Deep canonical correlation analysis. In Proceedings of the International Conference on Machine Learning, Miami, FL, USA, 4–7 December 2013; pp. 1247–1255.
40. Revaud, J.; Douze, M.; Schmid, C.; Jégou, H. Event retrieval in large video collections with circulant temporal encoding. In Proceedings of the 2013 IEEE/CVF Conference on Computer Vision and Pattern Recognition, Portland, OR, USA, 23–28 June 2013; pp. 2459–2466.
41. Wu, Y.; Ji, Q. Robust facial landmark detection under significant head poses and occlusion. In Proceedings of the 2015 IEEE International Conference on Computer Vision, Santiago, Chile, 7–13 December 2015; pp. 3658–3666.
42. Zhu, S.; Li, C.; Change Loy, C.; Tang, X. Face alignment by coarse-to-fine shape searching. In Proceedings of the 2015 IEEE/CVF Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, 7–12 June 2015; pp. 4998–5006.
43. Zhang, Z.; Luo, P.; Loy, C.C.; Tang, X. Learning deep representation for face alignment with auxiliary attributes. *IEEE Trans. Pattern Anal. Mach. Intell.* **2015**, *38*, 918–930. [CrossRef]
44. Ranjan, R.; Patel, V.M.; Chellappa, R. Hyperface: A deep multi-task learning framework for face detection, landmark localization, pose estimation, and gender recognition. *IEEE Trans. Pattern Anal. Mach. Intell.* **2017**, *41*, 121–135. [CrossRef]
45. Tompson, J.; Goroshin, R.; Jain, A.; LeCun, Y.; Bregler, C. Efficient object localization using convolutional networks. In Proceedings of the 2015 IEEE/CVF Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, 7–12 June 2015; pp. 648–656.
46. Papandreou, G.; Zhu, T.; Chen, L.C.; Gidaris, S.; Tompson, J.; Murphy, K. Personlab: Person pose estimation and instance segmentation with a bottom-up, part-based, geometric embedding model. In Proceedings of the European Conference on Computer Vision, Munich, Germany, 8–14 September 2018; pp. 269–286.
47. Ning, G.; Pei, J.; Huang, H. Lighttrack: A generic framework for online top-down human pose tracking. In Proceedings of the 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops, Seattle, WA, USA, 13–19 June 2020; pp. 1034–1035.
48. Burghardt, T.; Ćalić, J. Analysing animal behaviour in wildlife videos using face detection and tracking. *IEEE Proc.-Vision Image Signal Process.* **2006**, *153*, 305–312. [CrossRef]
49. Manning, T.; Somarriba, M.; Roehe, R.; Turner, S.; Wang, H.; Zheng, H.; Kelly, B.; Lynch, J.; Walsh, P. Automated Object Tracking for Animal Behaviour Studies. In Proceedings of the 2019 IEEE International Conference on Bioinformatics and Biomedicine (BIBM), San Diego, CA, USA, 18–21 November 2019; IEEE: Piscataway, NJ, USA, 2019; pp. 1876–1883.
50. Bonneau, M.; Vayssade, J.A.; Troupe, W.; Arquet, R. Outdoor animal tracking combining neural network and time-lapse cameras. *Comput. Electron. Agric.* **2020**, *168*, 105150. [CrossRef]
51. Vo, M.; Yumer, E.; Sunkavalli, K.; Hadap, S.; Sheikh, Y.; Narasimhan, S.G. Self-supervised multi-view person association and its applications. *IEEE Trans. Pattern Anal. Mach. Intell.* **2020**, *43*, 2794–2808. [CrossRef] [PubMed]
52. Jenni, S.; Favaro, P. Self-supervised feature learning by learning to spot artifacts. In Proceedings of the 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 2733–2742.
53. Zhang, R.; Isola, P.; Efros, A.A. Colorful image colorization. In Proceedings of the European Conference on Computer Vision, Amsterdam, The Netherlands, 8–16 October 2016; Springer: Cham, Switzerland, 2016; pp. 649–666.
54. Noroozi, M.; Favaro, P. Unsupervised learning of visual representations by solving jigsaw puzzles. In Proceedings of the European Conference on Computer Vision, Amsterdam, The Netherlands, 8–16 October 2016; Springer: Cham, Switzerland, 2016; pp. 69–84.
55. Pathak, D.; Krahenbuhl, P.; Donahue, J.; Darrell, T.; Efros, A.A. Context encoders: Feature learning by inpainting. In Proceedings of the 2016 IEEE/CVF Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 2536–2544.
56. Wang, X.; He, K.; Gupta, A. Transitive invariance for self-supervised visual representation learning. In Proceedings of the 2017 IEEE International Conference on Computer Vision, Beijing, China, 17–20 September 2017; pp. 1329–1338.
57. Zhao, R.; Xiong, R.; Ding, Z.; Fan, X.; Zhang, J.; Huang, T. MRDFlow: Unsupervised Optical Flow Estimation Network with Multi-Scale Recurrent Decoder. *IEEE Trans. Circuits Syst. Video Technol.* **2021**, *32*, 4639–4652. [CrossRef]
58. Sumer, O.; Dencker, T.; Ommer, B. Self-supervised learning of pose embeddings from spatiotemporal relations in videos. In Proceedings of the 2017 IEEE International Conference on Computer Vision, Beijing, China, 17–20 September 2017; pp. 4298–4307.

59. Gomez, L.; Patel, Y.; Rusiñol, M.; Karatzas, D.; Jawahar, C. Self-supervised learning of visual features through embedding images into text topic spaces. In Proceedings of the 2017 IEEE/CVF Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 4230–4239.
60. Owens, A.; Wu, J.; McDermott, J.H.; Freeman, W.T.; Torralba, A. Ambient sound provides supervision for visual learning. In Proceedings of the European Conference on Computer Vision, Amsterdam, The Netherlands, 8–16 October 2016; Springer: Cham, Switzerland, 2016; pp. 801–816.
61. Crawford, E.; Pineau, J. Spatially invariant unsupervised object detection with convolutional neural networks. In Proceedings of the 2019 AAAI Conference on Artificial Intelligence, Honolulu, HI, USA, 27 January–1 February 2019; Volume 33, pp. 3412–3420.
62. Rhodin, H.; Constantin, V.; Katircioglu, I.; Salzmann, M.; Fua, P. Neural scene decomposition for multi-person motion capture. In Proceedings of the 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 15–20 June 2019; pp. 7703–7713.
63. Katircioglu, I.; Rhodin, H.; Constantin, V.; Spörri, J.; Salzmann, M.; Fua, P. Self-supervised Training of Proposal-based Segmentation via Background Prediction. *arXiv* **2019**, arXiv:1907.08051.
64. Chen, L.C.; Papandreou, G.; Schroff, F.; Adam, H. Rethinking atrous convolution for semantic image segmentation. *arXiv* **2017**, arXiv:1706.05587.
65. Ronneberger, O.; Fischer, P.; Brox, T. U-net: Convolutional networks for biomedical image segmentation. In Proceedings of the 2015 International Conference on Medical Image Computing and Computer-Assisted Intervention, Munich, Germany, 5–9 October 2015; Springer: Cham, Switzerland, 2015; pp. 234–241.
66. Ionescu, C.; Papava, D.; Olaru, V.; Sminchisescu, C. Human3.6m: Large scale datasets and predictive methods for 3d human sensing in natural environments. *IEEE Trans. Pattern Anal. Mach. Intell.* **2013**, *36*, 1325–1339. [CrossRef]
67. Ionescu, C.; Li, F.; Sminchisescu, C. Latent structured models for human pose estimation. In *Proceedings of the 2011 IEEE International Conference on Computer Vision, Barcelona, Spain, 6–13 November 2011*; IEEE: Piscataway, NJ, USA, 2011; pp. 2220–2227.
68. Zhao, R.; Ouyang, W.; Wang, X. Unsupervised salience learning for person re-identification. In Proceedings of the 2013 IEEE/CVF Conference on Computer Vision and Pattern Recognition, Portland, OR, USA, 23–28 June 2013; pp. 3586–3593.
69. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep residual learning for image recognition. In Proceedings of the 2016 IEEE/CVF Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 770–778.
70. Kingma, D.P.; Ba, J. Adam: A method for stochastic optimization. *arXiv* **2014**, arXiv:1412.6980.
71. Baraldi, L.; Douze, M.; Cucchiara, R.; Jégou, H. LAMV: Learning to align and match videos with kernelized temporal layers. In Proceedings of the 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 7804–7813.
72. Sitzmann, V.; Martel, J.; Bergman, A.; Lindell, D.; Wetzstein, G. Implicit neural representations with periodic activation functions. *Adv. Neural Inf. Process. Syst.* **2020**, *33*, 7462–7473.