A Neuromorphic Approach to Object Detection and Recognition in Airborne Videos with Stabilization*

Yang Chen, Deepak Khosla, David Huber, Kyungnam Kim, and Shinko Y. Cheng

HRL Laboratories, LLC, Malibu, CA 90265

Abstract. Research has shown that the application of an attention algorithm to the front-end of an object recognition system can provide a boost in performance over extracting regions from an image in an unguided manner. However, when video imagery is taken from a moving platform, attention algorithms such as saliency can lose their potency. In this paper, we show that this loss is due to the motion channels in the saliency algorithm not being able to distinguish object motion from motion caused by platform movement in the videos, and that an object recognition system for such videos can be improved through the application of image stabilization and saliency. We apply this algorithm to airborne video samples from the DARPA VIVID dataset and demonstrate that the combination of stabilization and saliency significantly improves object recognition system performance for both stationary and moving objects.

1 Introduction

Object or target recognition in aerial videos has been a topic in machine vision research for many years. The traditional approach to this problem involves a two-step process: (1) detecting moving objects and tracking them over a certain number of video frames to select one or more regions of interest (ROI) in the frames, and (2) applying an object recognition algorithm on these ROIs, which may be bounding boxes or tight-fitting polygons. Unfortunately, this approach is limited in that it can only detect and recognize moving objects. Most applications with aerial videos involve both static and moving objects; thus, the use of both *form* and *motion* features is required to adequately detect all objects.

The brute-force solution to the recognition problem from a moving platform involves performing raster scan recognition over the entire frame so as to cover both static and moving objects, which suffers from a high processing load. Also, depending on the recognition method selected, it may be necessary to process the images at several scales (*e.g.*, HMAX [1,2]), further increasing the processing load. There is a need for fast and robust algorithms that detect potential ROIs with static and moving objects in aerial videos with high accuracy, which can then be processed by the

* This work was partially supported by the Defense Advanced Research Projects Agency NeoVision2 program (contract No. HR0011-10-C-0033). The views and conclusions contained in this document are those of the authors and should not be interpreted as representing the official policies, either expressly or implied, of the Defense Advanced Research Projects Agency or the U.S. Government.

G. Bebis et al. (Eds.): ISVC 2011, Part II, LNCS 6939, pp. 126–135, 2011.

recognition algorithm. An ideal algorithm is one that detects only ROIs corresponding to true objects (i.e., no false alarms), providing the downstream recognition algorithm with the maximum chance of success.

Neuromorphic attention algorithms, such as feature- or object-based saliency [3-7], can be used to find and extract regions of interest from video imagery. These algorithms process a scene and detect anomalies in its structure, such as sharp contrasts in color or intensity, strange geometries (such as a vertical element in a horizontally-dominated scene), or parts of the scene that appear to change with time (moving objects or things that appear to flicker) and return a result in the form of a "saliency map", which indicates how interesting or distinct a given region of the scene is.

Feature-based saliency algorithms process the scene pixel-by-pixel and find specific small regions that "stand out" against the rest of the scene. An example of this type of attention model is the NVT algorithm [3] and algorithms based on the Quaternion Fourier Transform [4] or spectral residual of the image [5]. This model of attention has often been described as a spotlight that focuses attention on a specific part of the scene without any concept of what it is actually highlighting. Typically, the spotlight is set to be some predetermined size that is larger than the expected object size, and the entire region is excised for further analysis.

An alternative to the feature-based saliency algorithm is the object-based approach, which attempts to extract entire objects from the scene based on continuous expanses of salient features. Like the feature-based approach, these algorithms process an image and extract regions that stand out from the rest of the scene. However, instead of acting like a spotlight, these algorithms employ the feature maps as a means to determine the object boundary. Consequently, this approach is able to segment complete objects from the scene. Examples of object-based saliency algorithms are the work of Orabona, *et al*. [6] and Huber and Khosla [7].

It has been previously shown that employing an attention algorithm as a front-end to a recognition system can dramatically improve object recognition results, both through increased correct detections and lower false alarms [8-10] when the camera is stationary. In this instance, an attention algorithm is applied to the frames in a video sequence and regions of interest (ROI) are extracted based on their saliency, which are used as cues and fed into the object recognition algorithm. By combining a biologically-inspired attention algorithm, which can detect both moving and stationary objects, with a biologically-inspired recognition algorithm, one can form a powerful visual recognition engine without going through the traditional detect-and-track paradigm. This permits the detection and recognition of both moving and stationary objects at higher speed than with traditional approaches.

However, current attention algorithms are only effective in stationary scenes; saliency maps obtained from a moving platform, as is the case with aerial videos, often contain a great deal of noise and produce a large number of "false alarms" corresponding to background features that do not correspond to objects in the scene. These errors are likely due to the egomotion of the camera conflicting with the motion detection of the saliency algorithm. Our analysis shows that these algorithms cannot differentiate between camera motion and object motion in the scene. This is a severe limitation in the application of saliency as a front-end for object recognition systems, since much surveillance video is obtained from moving aerial platforms. In light of

the improvement in the results in [8], it is critical that a method of computing saliency on moving platforms be developed.

In this paper we describe an architecture that performs object recognition in videos from a moving platform, and can detect both moving and stationary objects by using bio-inspired attention and recognition algorithms. We preprocess the aerial videos with video stabilization, which allows the images of the ground objects to be easily detected as salient points by the attention algorithm without suffering from motion-induced clutter. We extract an image chip (*i.e.*, ROI), which can be a fixed size bounding box or a tight-fitting object shape computed using the same features [10], and apply a bio-inspired object recognition algorithm. We demonstrate that this architecture significantly improves performance in terms of recognition rate/false alarm metric as validated on VIVID aerial video dataset.

2 Method

For this work, we employ a three-stage approach to object recognition, which is discussed in detail in this section. First, we apply a video stabilization function, which finds the spatial transformation that can be used to warp video images in neighboring frames into a common coordinate system and eliminate the apparent motion due to sensor platform movement. Next, we apply a neuromorphic attention algorithm to the stabilized video images and produce a set of locations in the images that are highly likely to contain objects of interest. The bio-inspired feature extraction function takes a small image chip (i.e., the ROI) around each salient point and extracts high-dimensional feature vectors based on models developed following human visual cortex. These features are used by the classification engine that employs an algorithm such as a Support Vector Machine (SVM), to either classify the features into an object class or reject the image chip.

2.1 Video Stabilization

The purpose of video stabilization is to compensate the motion in the video images caused by the motion of the camera and/or its platform. Our method of image stabilization consists of four steps, feature detection, matching, image transformation estimation and image warping.

We use the Scale Invariant Feature Transform (SIFT) as feature descriptor, which is invariant to scale, orientation, and affine distortions, to extract key points for the image. Key points are defined as maxima and minima of the result of difference of Gaussians function applied in scale-space to a series of smoothed and re-sampled images. Dominant orientations are assigned to localized key points. SIFT feature descriptors are 128-dimensional vectors representing the gradient orientation histograms and can be used to compare if two image key points are similar (*i.e.*, they are from the same point in the scene).

Feature matching compares the two sets of SIFT features and match the key points from one image to the other that have similar SIFT features. This results in a list of candidate set of matching points from the two images to be filtered in the next step. A match for a key point in one image is defined as the key point in the other image with the minimum Euclidean distance based on the descriptor vectors of the key points.

The list of matching points obtained this way is not very reliable in that incorrect matches can happen due to noise and inherence capability of SIFT descriptor in distinguishing certain type of key points. To achieve more reliable matching, we apply RANSAC (Random Sample Consensus) which is an iterative method to estimate parameters of a mathematical model from a set of observed data that contains outliers. We use RANSAC to find a homography transform (model) that fits the candidate set of matches. As a result we get a set of correct matches as well as an accurate transformation (homography) between the two images.

The final step in video stabilization is to warp the image frames into a global coordinate frame so that the warped images show no platform induced image motion. In a "blocked" mode of operation, we choose a block size of N frames in which each frame is warped to the first frame in the block using the homography transformation found as described above (e.g., frames 1, …, N are warped into the coordinate system of frame 1; frames N+1, …, 2N are warped into frame N+1, and so forth). This way, the images within each block are stabilized with respect to the first frame of the block, while the images between blocks are not stabilized. Alternatively, in a "non-blocked" mode of operation, we warp the previous image frame for every new input frame (the current frame) so that the pair of current and previous images are always registered for the attention algorithm. Both approaches allow camera motion without having to maintain a large global image frame buffer. In our experiments, we produced the stabilized versions of our input aerial videos in block mode with a block size of 10. The block size should be determined by the video frame rate and the platform speed and altitude. Our videos were taken at 30 fps (altitude = 800–1200 meters; speed = 40-70 meters/sec). If the scene doesn't change much, one can use larger block sizes. Otherwise, the block size should be smaller to ensure proper overlap among the images in the same block.

2.2 Neuromorphic Attention for Object Detection

Following video stabilization, we apply a bio-inspired visual attention algorithm similar to [7] to detect locations in the video images that are likely to contain objects of interest. While the literature is rich with different methods (*e.g.*, [3-7]), most saliency algorithms work in the same basic manner: accepting two consecutive frames as input at any given time and outputting a saliency map, which indicates how "interesting" a given location in the frame is relative to its surroundings. This is done by assigning a score to each spatial location in the image that measures its variance from the rest of the image. Saliency algorithms generally contain one module for static object detection and another for finding moving objects.

For static detection, the image data for the current frame is decomposed into channels that correspond to color and intensity; red, blue, green, yellow, and luminance are commonly used, which are processed as opposing pairs with a positive "center" receptive field of one color and a negative "surround" receptive field of its opponent color. This center-surround color opponency mimics the processing of the mammalian visual system and allows the system to find strong color opposition in the scene. Color opponency maps are computed for the red/green and blue/yellow pairings by performing the convolution of each color channel with a narrow-band "center" Gaussian kernel and a wide-band "surround" Gaussian kernel. Each surround result is subtracted from its appropriate center result for each color pairing, providing

four color opponency maps: redC/greenS, greenC/redS, blueC/yellowS, and yellowC/blueS. Similarly, center-surround maps for orientation and intensity are computed by convolving the luminance channel with narrow- and wide-band Gabor and Gaussian filters, respectively. The orientation channel detects geometrical anomalies, such as a single horizontal element in a primarily vertical field, while the intensity channel picks up spots of dark or light against an opposing backdrop. Because these features employ a single frame, the motion of the platform is likely to have little effect on the results.

Motion processing in a saliency algorithm is carried out by computing the difference between the intensity channels for two consecutive frames for offset in five directions (up, down, left, right, and in-place, or zero-offset). These channels detect change between the two frames and pick up on motion or, in the case of the zero-offset channel, what appears to be flickering of light in the scene. Because these channels use a pair of frames for processing, scenes from a moving platform can cause these channels to provide spurious or false results due to the algorithm confusing stationary features that appear to move with actual moving objects.

A saliency map is constructed from the weighted contributions of the four color opponency maps, the intensity map, the four orientation maps, and the five motion maps by a sequence of addition and normalization of maps that correspond to common features. For object recognition, we extract the peaks from the saliency map that the algorithm returns, obtaining a list of locations in the image. In theory, these are the regions that the human eyes are likely to attend to that correspond to objects of interest. The peak threshold is set sufficiently low that all possible targets are detected, (*i.e.*, no false negatives). We seed the visual recognition engine with the image chips or ROIs (128x128 regions extracted from the image) that are centered at these peaks.

2.3 Biologically-Inspired Visual Recognition

HMAX (or CBCL) is a feed-forward model of mammalian visual cortex [1, 2] that has been validated to perform similarly as humans do in fast object recognition tasks. At the heart of this model is hierarchy of alternating layers of filters simulating simple and complex cells in the mammalian visual cortex. The simple cells perform template matching, while the complex cells perform max-pooling and subsampling, which achieves local invariance to shift. As the algorithm moves to the higher layers, the features become more invariant with a wider receptive field. At the top layer, this model outputs a vector of high-dimensional features typically ranging in size from hundreds to a few thousand elements that can be used to classify the input image from the bottom layer.

In our experiments, we used a model similar to that described in Mutch and Lowe [11], but with a base image size of 128x128 and 8 layers of image pyramid. 200 random C1 patches were used, which are sampled from a set of training images of similar scenes as in our aerial video images. This results in a feature vector of 200 dimensions for each input image of 128x128.

To complete the HMAX/CBCL based visual recognition engine, a set of labeled training images that includes both objects of interest and background clutter are presented to the HMAX/CBCL model and the resulting feature vectors are used to

train a Support Vector Machine (SVM) classifier. Once trained, the SVM classifier can be operated on-line in the system to provide image classification (such as a vehicle, bike, pedestrian or background) with a confidence value. We employ the SVM classifier as a convenience which also has been proven to perform well for a variety of classification tasks. However, any multi-class classification method that can handle high-dimensional feature would be sufficient.

3 Results and Discussion

We validated the stabilization-saliency methodology that we present here using a combination of CPU/GPU implementations of the three modules discussed in Section 2. The algorithms were applied to the DARPA VIVID dataset, which consists of a series of color videos taken from a moving aerial platform (Figure 1). There are a number of object types present in these videos, including vehicles (cars and trucks), motorcycles, and pedestrians. In each video, potential objects can be in plain view or partially occluded; in most cases the objects are moving. For our experiments, we first ran the base-line system that involves the application of saliency without video stabilization. We trained the HMAX/CBCL and the SVM classifier using sample object images from a set of 6 training videos, each containing between 1800 and 1900 frames and tested on a different set of 6 videos than those used in training. Then we retested the system with the same test data after stabilizing the videos in blocked mode with block size N=10.

Fig. 1. Sample images from DARPA VIVID Dataset. This dataset contains color videos of various objects such as vehicles, motorcycles, and pedestrians at 640x480 resolution.

Our first objective was to determine the specific reasons that the saliency algorithm perform poorly on videos from a moving platform. We ran the saliency algorithm on the unstabilized VIVID videos and saw a significant drop in object detection performance over what we would have expected if the video were shot from a stationary camera. Figure 2 curve (a) shows the receiver operating characteristic (ROC) curve for this trial, and illustrates the probability of object detection (Pd) as a function of false positives per image (FPPI). Here Pd is defined as the ratio of number of salient chips (section 2.2) having non-zero intersections with the target

bounding boxes to the number of ground truth targets (regardless of class) in each image, averaged over all images in the test sequences. False positives are those salient chips that do not intersect with any target bounding boxes. FPPI, an average over all image frames, is used instead of traditional false positive rate (or FAR) because FPPI directly translates to the number of false positives per unit time given the video frame rate, which is a preferred measurement of false alarms for image analysts.

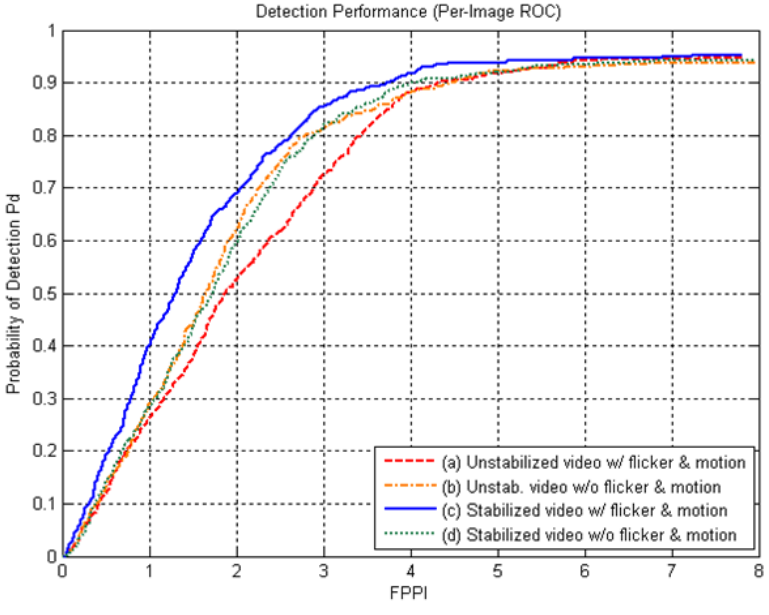

Fig. 2. Object detection performance based on saliency with and without motion and flicker channels for sample videos from VIVID data set. (a) The saliency algorithm performs poorly on unstabilized videos when motion and flicker channels are used by the saliency algorithm. (b) When flicker and motion channels are not used, the performance of saliency is restored to certain extent. (c) When the video is stabilized, the full saliency algorithm achieves the best performance. (d) When motion channels are not used, saliency performance on stabilized videos is similar to that on unstabilized videos. The horizontal axis indicates the false positive per image (FPPI) (see text for explanation).

Suspecting that the algorithm was picking up extraneous saliency signatures from the egomotion of the camera (*i.e.*, frame to frame motion due to camera motion boosted certain image features to have unusually high saliency scores), we ran the trial again with the motion channels disabled and saw a significant increase in performance (Figure 2, curve (b)), though not as good as the full saliency algorithm from a stationary camera. This clearly shows that the motion channels are rendered impotent by the image motion due to platform movement, and the overall detection results suffer as a consequence of false alarms that effectively swamp the other feature maps (e.g., intensity, color, orientation). This is likely due to the way that the saliency algorithm processes motion. By differencing the intensity maps of consecutive frames, the

saliency algorithm detects motion by changes in the intensity patterns of the image frames in various directions over time. However, this method only works locally and does not notice bulk, consistent motion of all objects in a frame caused by a moving camera. Therefore, the saliency algorithm cannot differentiate between a moving object viewed by a stationary camera and a stationary object viewed by a moving camera because all it sees are blobs that appear to move within the frame. By removing the motion channels from the saliency calculation, we eliminate a major source of noise, which provides the observed marginal improvement in the probability of detection.

From this preliminary analysis, we are able to infer that the moving platform ultimately leads to the loss of the effectiveness of the motion channels in the saliency algorithm. Since the motion processing in a saliency algorithm works on pairs of consecutive images, they should be stabilized with respect to one another prior to processing by saliency; a method of image stabilization that takes care to make the images look stationary to the saliency algorithm is a likely solution to this problem.

We applied the stabilization method described in 2.1 to the same VIVID videos and repeated the trials for the saliency algorithm with and without motion channels. These results are displayed as curves (c) and (d) in Figure 2. The benefit from stabilizing the image is immediately apparent; this result provides a large benefit over its unstabilized analogue. However, what is interesting is how closely the results for the saliency on the stabilized and the unstabilized videos without using motion components correlate with one another. This indicates that the static components of saliency algorithm behave nearly identically in both cases and it validates the hypothesis that the motion channels suffer when the video frames are not stabilized, which degrades the system performance.

Fig. 3. Comparison of salient points using motion unstabilized (*left*) and stabilized (*right*) videos. The moving camera picks up on spurious high contrast areas on the ground (*left*), which disappear (*right*) when the video is stabilized prior to saliency processing.

Figure 3 shows the ROI provided by the saliency algorithm in unstabilized (left) and stabilized (right) input videos. All regions that exceed a given threshold in each image are defined as ROIs and denoted by a box. As can be seen, the most salient objects in the stabilized scene all correspond to vehicles, whereas the ROIs in the unstabilized video are more dispersed due to the platform motion. This validates our

hypothesis that the saliency algorithm is swamped by spurious motion signals when the camera is moving that prevent actual moving targets from being detected. In this case, patches of light-on-dark contrast on the ground appear to move in the unstabilized imagery, which produce a stronger saliency signal than the moving vehicles in the scene (due to higher overall contrast). However, when the scene is stabilized prior to applying the saliency algorithm, these patches no longer appear to move and saliency is able to detect the vehicles.

To quantify the benefits of better target detection performance to the final object recognition system performance, next we ran the classifier on the salient chips provided for the stabilized and unstabilized VIVID videos and summarized the results as ROC curves (Figure 4). Here the SVM classifier was trained on 3 target classes (vehicle, bike and pedestrian) plus the background class using samples from the 6 training sequences and applied to the salient chips from the test sequences.

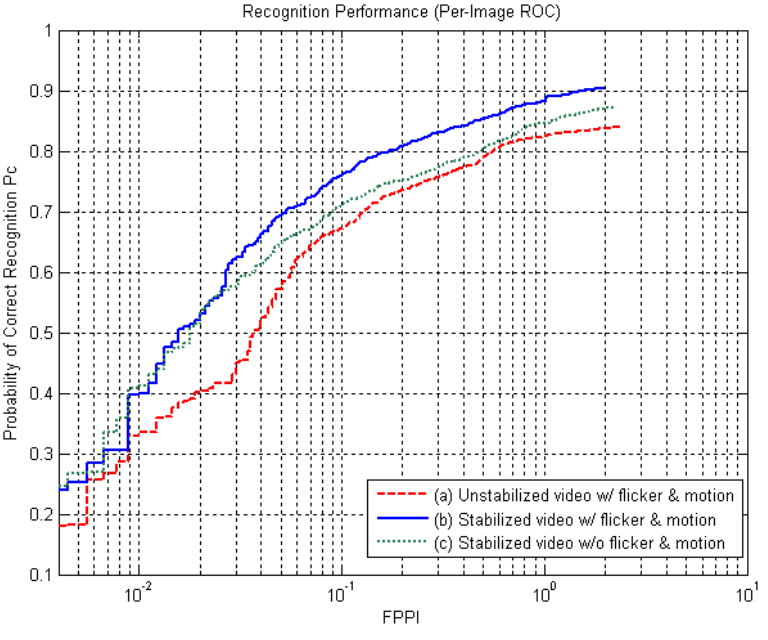

Fig. 4. Performance of the HMAX/CBCL-SVM based object recognition system with and without video stabilization. (a) System with unstabilized video based on ROIs provided by full saliency; (b) Stabilizing the video greatly improves the recognition system performance; (c) even when flicker and motion channels are not used by the saliency algorithm, video stabilization can still boost overall system performance. The horizontal axis measures the false positive per image (FPPI) (see text for Figure 2. for explanation).

As can be seen from Figure 4, the system with video stabilization performs much better than it does without video stabilization (the performance is better if the ROC is towards the top and left, meaning higher recognition rate and lower false alarms). This shows that the better detection performance shown in Figure 2 translates to performance benefits in object recognition of the overall system.

4 Conclusion

The application of a saliency algorithm as a front end to an object recognition system can improve overall system performance. However, this advantage is greatly compromised when the camera used to capture the video is attached to a moving platform, due to image motion caused by platform movement. In fact, the motion processing portion of the saliency algorithm is not only wasted, but also harmful to system performance. We have shown in this paper that employing an image stabilization process prior to the application of the saliency algorithm can restore the effectiveness of the motion channels of the saliency algorithm and achieve a significant improvement in performance for object detection and recognition. Furthermore, as a practical guideline, when video stabilization is unavailable or infeasible to implement, saliency algorithm works better if the motion channels are disabled in the saliency algorithm.

References

1. Serre, T., Poggio, T.: A Neuromorphic Approach to Computer Vision. Communications of the ACM (online) 53(10), 54–61 (2010)
2. Serre, T., Wolf, L., Bileschi, S., Riesenhuber, M., Poggio, T.: Robust Object Recognition with Cortex-Like Mechanisms. IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI) 29(3), 411–426 (2007)
3. Itti, L., Koch, C.: A Saliency-Based Search Mechanism for Overt and Covert Shifts of Visual Attention. Vision Research 40, 1489–1506 (2000)
4. Guo, C., Ma, Q., Zhang, L.: Spatio-temporal Saliency Detection Using Phase Spectrum of Quaternion Fourier Tranform. In: Proc. CVPR, pp. 1–8 (2008)
5. Hou, X., Zhang, L.: Saliency Detection: A Spectral Residual Approach. In: Proc. CVPR, pp. 1–8 (2007)
6. Orabona, F., Metta, G., Sandini, G.: A Proto-object Based Visual Attention Model. In: Paletta, L., Rome, E. (eds.) WAPCV 2007. LNCS (LNAI), vol. 4840, pp. 198–215. Springer, Heidelberg (2007)
7. Huber, D., Khosla, D.: A Bio-Inspired Method and System for Visual Object-Based Attention and Segmentation. In: Proc. SPIE DSS, vol. 7696 (2010)
8. Chikkerur, S., Serre, T., Poggio, T.: Attentive Processing Improves Object Recognition. Massachusetts Institute of Technology Technical Report: MIT-CSAIL-TR-2009-046 (2009)
9. Rutishauser, U., Walther, D., Koch, C., Perona, P.: Is Bottom-Up Attention Useful for Object Recognition? In: Proc. CVPR, vol. 2, pp. 37–44 (2004)
10. Walther, D., Koch, C.: Modeling Attention to Salient Proto-Objects. Neural Networks 19, 1395–1407 (2006)
11. Mutch, J., Lowe, D.: Multiclass Object Recognition with Sparse, Localized Features. In: Proc. CVPR, pp. 11–18 (2006)