Automatic Play Segmentation of Hockey Videos

Hemanth Pidaparthy
York University
phemanth@eecs.yorku.ca

Michael H. Dowling
Queens University
michael.dowling@queensu.ca

James H. Elder
York University
jelder@yorku.ca

Abstract

Most team sports such as hockey involve periods of active play interleaved with breaks in play. When watching a game remotely, many fans would prefer an abbreviated game showing only periods of active play. Here we address the problem of identifying these periods in order to produce a time-compressed viewing experience. Our approach is based on a hidden Markov model of play state driven by deep visual and optional auditory cues. We find that our deep visual cues generalize well across different cameras and that auditory cues can improve performance but only if unsupervised methods are used to adapt emission distributions to domain shift across games. Our system achieves temporal compression rates of 20-50% at a recall of 96%.

1. Introduction

Automation of sports videography has the potential to provide professional-level viewing experiences at a cost that is affordable for amateur sport. Autonomous camera planning systems have been proposed [2, 3, 9], however, these systems deliver continuous video over the entire game. Typical amateur ice hockey games feature between 40 and 60 minutes of actual game play. However, these games are played over the course of 60 to 110 minutes, with downtime due to the warm-up before the start of a period and the breaks between plays when the referee collects the puck and the players set up for the ensuing face-off. Also, there is a 15 minute break between periods for ice re-surfacing. Automatic identification of these stoppages would allow abbreviation of the video.

We propose a novel system that uses visual cues from a single wide-field camera and optional auditory cues. We explore two different visual cues. The first is based on the optic flow - players tend to move faster during play than breaks. However, we find that this cue is fallible: motion on the ice can sometimes be substantial during breaks and sometimes quite limited during periods of play. This motivates the development of a more complex deep visual classifier that takes not only the optic flow but also the RGB image and detected player positions as input.

We also explore the utility of auditory cues - more specifically, the referee whistle that starts and stops play. While not directly informative of the current state, the whistle does serve to identify the timing of state transitions, and thus can potentially contribute to performance.

To take into account temporal dependencies, we employ a hidden Markov model (HMM), which, while simplifying modeling through conditional independence approximations, allows (1) optimal probabilistic integration of these noisy cues and (2) an account of temporal dependencies captured through the state transition matrix.

We also propose a method for unsupervised domain adaptation of the HMM, iteratively updating emission and/or transition probability distributions at inference, using the predicted state sequence, and show that this is critical to benefitting from auditory cues.

In summary, the main contributions of this paper are: 1) a novel probabilistic framework for classifying video sequences into periods of play or no-play and 2) a novel approach to handling auditory domain shift that leads to improved performance from multisensory integration.

2. Prior Work

Several approaches have been proposed to address play-break segmentation. Some use play-break segmentation for automatic highlight generation [4, 14] or event detection [13], while others use event detection to guide play-break segmentation [1]. Some of the initial methods for play-break segmentation used rule-based approaches that combined text graphics on a broadcast feed with audio cues from the crowd and commentator [15] or the type of broadcast camera shot [4, 14, 17, 13]. Ekin *et al.* [4] and Tjondronegoro *et al.* [14] selected the play frames as key frames for automatic highlights from broadcast soccer videos, while Tavassolipour *et al.* [13] used the play frames to detect events in broadcast soccer videos.

All of these approaches used broadcast cues (camera shot type) or production cues (graphics and commentary) for play-break segmentation, and thus are not directly relevant to unedited amateur sport video recorded automatically

with fixed cameras. We are aware of only one prior paper that addresses a similar problem for unedited videos. Carbonneau *et al.* [1] trained an SVM on Bag-of-Words features in ice hockey videos to detect key events such as line changes, face-offs and preliminary play-breaks. In a second stage, these events are integrated with spatio-temporal features to segment the video into play and non-play intervals. The method was trained and evaluated on disjoint intervals of a single hockey game recorded by two different cameras.

Our research goes beyond this prior work in several important ways. First, our approach classifies frames as play and no-play without requiring the detection of finer-grain events like line changes. Second, we show how the temporal dependencies between states can be captured and integrated with these probabilistic cues within an HMM framework that allows maximum a-posteriori (MAP) or minimum-loss solutions to be computed in linear time. (While HMMs have previously been used for broadcast video [17, 13] they have not been applied to unedited video.) Third, we introduce a novel method for handling auditory domain shift that is critical for integration with visual cues. Finally, we go beyond this prior work by showing generalization across games, rinks and viewing parameters, which is critical for successful deployment.

3. Dataset

As there are no public datasets of multiple hockey games recorded with fixed cameras, we have created and labeled our own, which we will make public at www.elderlab.yorku.ca/resources. The dataset consists of 12 amateur hockey games recorded using three different high-resolution 30fps camera systems, placed in the stands, roughly aligned with the centre line and about 10m from the closest point on the ice:

Camera 1. Four games were recorded using a 4K Axis P1368-E camera (Fig 1a).

Camera 2. Five games were recorded using a proprietary system consisting of two 4K IP cameras with inter-camera rotation of 75 deg (Fig 1b). Nonlinear distortions were removed using a standard calibration procedure [18] and a template of the ice rink (Fig 2b) was employed to manually identify homographies between the two sensor planes (Fig 2a) and the ice surface. Finally, these homographies were used to reproject both cameras to a virtual cyclopean camera bisecting the two cameras, where the two images were stitched using a linear blending function (Fig 2c).

Camera 3. Three games were recorded using a 4K wide-FOV GoPro 5 camera (Fig 1c-1d), which also recorded synchronized audio at 48kHz.

The 9 games recorded by Cameras 1 and 2 and Games 2 and 3 recorded by Camera 3 were all played at the same rink, while Game 1 recorded by Camera 3 was played at a different rink. All games recorded by Cameras 1 and 2 were captured in their entirety, each game lasting from 1h28m - 2h30m, while only the first two of three periods in each game were captured by Camera 3 (35m-57m).

Camera 1 and Camera 2 were placed roughly 8 m and Camera 3 roughly 7m bove the ice surface. The substan-

(a) Camera 1

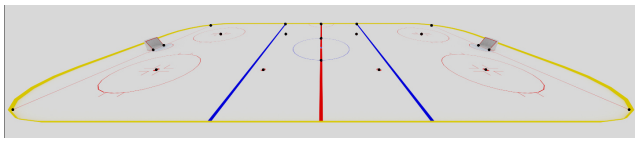
(b) Camera 2

(c) Camera 3 Game 1 Period 1

(d) Camera 3 Game 2 Period 1

Figure 1: Example frames from the dataset, after correction for radial distortion and masking of the non-rink regions.

(a) Images from the two cameras



(b) Template image

(c) Stitched image

Figure 2: Camera 2 view construction. (a) Example frames from two cameras. (b) Rink template. (c) Stitched image with homography control points.

tial radial distortion in all the videos was corrected using a standard calibration procedure [18]. To assess generalization over camera parameters, we varied the roll and tilt of Camera 3 by roughly ± 5 deg between games and periods (Fig 1c-1d). We close-cropped all videos to the rink after recording; due to variation in camera roll, two different vertical crops were employed for Camera 3, depending upon the game and period. See Table 1 for final resolutions.

Camera	Horizontal	Vertical
1	2,758	500
2	5,930	1,080
3	3,550	630-780

Table 1: Cropped video resolutions.

3.1. Ground-truthing

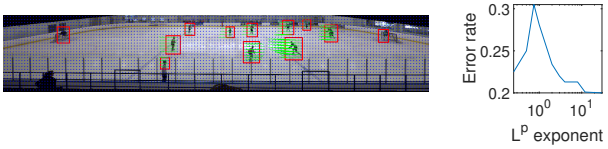
We manually ground-truthed all 12 games by marking the start and end of play intervals. For Cameras 1 and 2, we marked the start of play as the time instant when the referee dropped the puck during a face-off and the end of play by when the referee was seen to blow the whistle. Since we had audio for Camera 3, we instead identified state changes by the auditory whistle cue, marking both the beginning and end of whistle intervals, which were found to average 0.73 sec in duration.

We generally train and evaluate within camera systems, but also show that our deep visual cues generalize well across different camera systems as well as modest variations in extrinsic camera parameters, in the case of Camera 3. For all 3 camera systems, training and evaluation is performed on different games, using leave-one-game-out k-fold cross-validation.

4. Perceptual Cues - Visual

4.1. Maximum Optic Flow

We used the OpenCV implementation of Farneback's dense optic flow algorithm [5] and selected the flow fields lying within bounding boxes of players detected using a Faster-RCNN detector [11], fine-tuned on 3 games recorded using Camera 2 that were not part of this dataset (Fig 3a). Motion energy is generally higher during periods of play than during breaks, but given the sparse nature of the flow it is not immediately clear how to optimally aggregate the flow signal to create the strongest classifier. To explore this, we assessed a range of L^p norms over the optic flow vector magnitudes for Game 1 recorded using Camera 3, measuring classification error for distinguishing play from no-play states (Fig 3b). Interestingly, we found that error rate was lowest for very high exponents, which leads to a very simple and computationally efficient visual cue: the L^∞ norm of the optic flow, i.e., the maximum flow vector magnitude within detected player boxes.



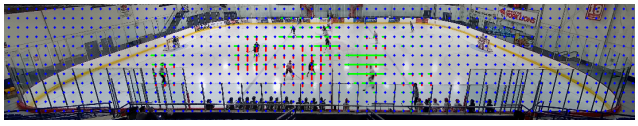
(a) (b)

Figure 3: (a) Example optic flow field within bounding boxes of detected players. (b) Error rate as a function of the L^p exponent used to aggregate the optic flow field.

4.2. End-to-End Deep Visual Classifier

We designed a small deep classifier to allow end-to-end training for play/no-play classification. (For Camera 3, whistle frames were included in the play intervals.) The 6 channels of input (Fig 4) consisted of a) the RGB image, b) horizontal and vertical optic flow maps and c) binary player position mask. All feature maps were normalized to have zero mean and unit variance, resized to 150 $\times$ 60 pixels, and then stacked to form a 6-channel input. The training dataset was augmented by left-right mirroring.

(a) RGB image



(b) Horizontal (green) and vertical (red) optic flow

(c) Binary player position mask

Figure 4: Input to deep visual play/no-play classifier.

We used PyTorch to develop the network, which consisted of two *conv-pool* modules followed by two fully connected layers - details are shown in Fig 5. Dropout was used between every fully connected layer. The output from the network was the softmax probability of the frame belonging to play or no-play classes. Cross-entropy loss between the predicted class and ground truth class was minimized using an SGD optimizer. The model was trained for 20 epochs with an initial learning rate of 0.01 and weight decay of 0.01. The learning rate was decreased by 50% every 5 epochs.

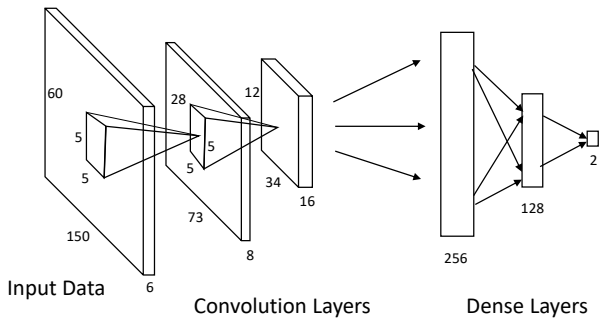


Figure 5: CNN trained to distinguishing between 'play' and 'no-play' frames.

We trained a separate model for each camera. For Cameras 1 and 2, one game was used for validation and one for test, and the remaining games used for training. For Camera 3, one game was used for test, one period from one of the other games was used for validation, and the remaining data were used for training.

4.3. Visual Cue Evaluation

We compared our two visual classifiers against two baseline deep classifiers trained to use as input the 512-dimensional output from the final fully connected layer of the ImageNet-trained ResNet18 network [6]. The first classifier consisted of two fully connected layers of dimensions 128 and 64, followed by a play/no-play softmax layer. The learning rate for this network was 0.001, weight decay was 0.01 and it was trained for 10 epochs. The second classifier was an SVM using an RBF kernel.

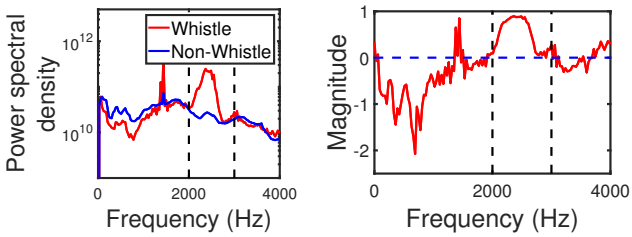
Table 2 shows the performance of the four visual classifiers. Across all cameras, the best performance was obtained using our end-to-end trained deep visual classifier.

	AUC scores		
	Camera 1	Camera 2	Camera 3
Resnet18 + FC	0.923 ± 0.018	0.907 ± 0.052	0.598 ± 0.03
Resnet18 + SVM	0.884 ± 0.009	0.844 ± 0.014	0.545 ± 0.01
Maximum optic flow	0.885 ± 0.011	0.818 ± 0.008	0.799 ± 0.028
End-to-end deep classifier	**0.977 ± 0.004**	**0.966 ± 0.005**	**0.819 ± 0.053**

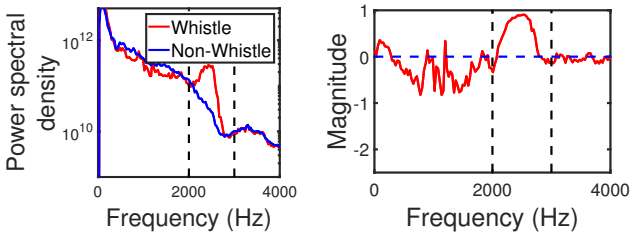
Table 2: AUC scores for visual play/no-play classification.

5. Perceptual cues - Auditory

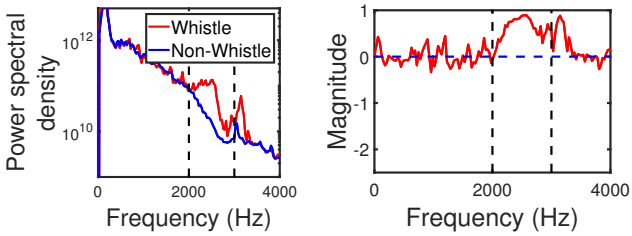
In ice hockey, referees blow their whistles to start and stop play. The whistle therefore can serve as an indicator of transitions between the play state and no-play state. For Camera 3, we partitioned the audio signal into 33 msec intervals, temporally aligned with the video frames. Since the audio was sampled at 48 kHz, each interval consisted of 1,600 samples. Fig 6 shows the power spectral density (PSD) averaged over whistle and non-whistle intervals for the three games recorded using Camera 3. These plots reveal several important facts. First, the overall volume of sound varies widely from game to game: While Game 1 is relatively quiet, Games 2 and 3 are quite noisy, with a lot of power in the low frequencies. Second, most of the whistle power lies in the 2-3 kHz range, however that power is not distributed evenly and the power of that signal and hence the signal-to-noise ratio varies widely from game to game.



(a) Game 1



(b) Game 2



(c) Game 3

Figure 6: Spectral analysis of whistle and non-whistle intervals. Left: spectral densities of whistle and non-whistle intervals. Right: Wiener filters.

To form a decision variable for each interval, we considered two candidate detectors:

Bandpass filter. We compute the integral of the PSD over the 2-3 kHz band. This is probabilistically optimal if both the signal and noise are additive, stationary, white Gaussian processes and the PSDs are identical outside this band (see below).

Wiener filter. Fig 6 shows that in fact the signal and noise are not white. Relaxing the condition that the PSDs be white and identical outside the 2-3 kHz band, for longer intervals (many samples), it can be shown ([8], Section 5.5) that probabilistically near-optimal detection is achieved by taking the inner product of the stimulus PSDs with the Wiener

filter

$$H(f) = \frac{P_{ss}(f)}{P_{ss}(f) + P_{nn}(f)} \quad (1)$$

where $P_{ss}(f)$ and $P_{nn}(f)$ are the PSD of the signal (whistle) and noise, respectively, as a function of frequency f.

In our case, we do not have direct knowledge of the whistle and noise PSDs and so must estimate them from the training data:

$$P_{ss}(f) \approx P_W(f) - P_{NW}(f) \quad (2)$$
$$P_{nn}(f) \approx P_{NW}(f) \quad (3)$$

where $P_W(f)$ and $P_{NW}(f)$ are the average PSDs over whistle and non-whistle training intervals, respectively. Thus we have that

$$H(f) \approx \frac{P_W(f) - P_{NW}(f)}{P_W(f)} \quad (4)$$
$$= 1 - \frac{P_{NW}(f)}{P_W(f)} \quad (5)$$

Fig 6 (right panels) show the resulting Weiner filters $H(f)$ estimated for each of the three games recorded by Camera 3. The filter is largely positive in the 2-3 kHz range but can become negative outside this range. This suggests that in fact the signals are not exactly stationary and/or additive. Two possibilities are that some acoustic signals are more likely to occur in non-whistle than in whistle intervals, and that, when the whistle is blown, auto-gain circuitry in the camera attenuates energy outside the whistle band.

To handle these deviations from our assumptions, we evaluated three versions of the Wiener filter:

1. **Wiener filter 1.** Take the inner product of the stimulus with the estimated Wiener filter over the entire frequency range, including negative values.

2. **Wiener filter 2.** Take the inner product of the stimulus with the rectified Wiener filter (negative values clipped to 0).

3. **Wiener filter 3.** Take the inner product of the stimulus with the rectified Wiener filter (negative values clipped to 0), only over the 2-3 kHz range.

Table 3 shows the average AUC scores for these four detectors using three-fold cross-validation on the three games recorded using Camera 3. Overall, the Wiener filter 3 detector performs best. Its advantage over the bandpass filter presumably derives from its ability to weight the input by the non-uniform SNR within the 2-3 kHz band. Its advantage over the other two Wiener variants probably reflects the inconsistency in the PSD across games outside this band.

	AUC score
Bandpass filter	0.919 ± 0.039
Wiener filter 1	0.779 ± 0.105
Wiener filter 2	0.809 ± 0.093
Wiener filter 3	0.943 ± 0.028

Table 3: The average cross-validated AUC score of the four whistle detectors for Camera 3.

6. Hidden Markov Model

Visual cues are seen to be useful for classifying video frames individually as play/no-play and auditory cues are useful for detecting the whistle. In order to put these cues together and reliably excise periods of non-play from the entire video, we need a model that captures statistical dependencies over time. Fig 7 shows an example of how the visual maximum optic flow and auditory cues vary over time within each game state.

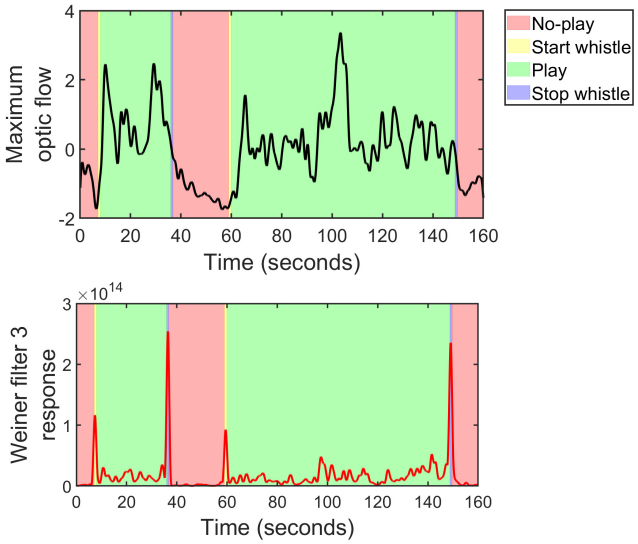
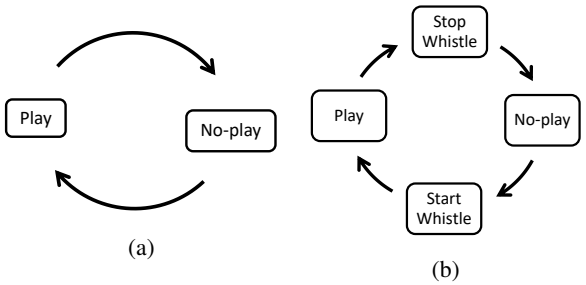


Figure 7: Visual and auditory cues for an example video segment from Camera 3 Game 1.

To capture these statistical dependencies, we employ a hidden Markov model (HMM) [10] of play state. For Cameras 1 and 2 (visual only), we employ a 2-state model (play/no-play) (Fig 8a). For Camera 3 (with audio), we employ a 4-state model that includes start and stop whistle states (Fig 8b). Table 4 shows the state transition probabilities learned from the labelled data.

In addition to the state transition probabilities, we need emission distributions for the observed visual and auditory cues, which we will treat as conditionally independent. We model all densities using Gaussian kernel density estimation with bandwidth selected by Silverman's rule [12]. Fig 9 shows these conditional distributions for one game from each camera and for our two visual cues: the maximum



(a) (b)

Figure 8: State transition graphs for (a) 2-state and (b) 4-state HMMs.

Camera	Transition	Probability
1	No-play→Play	0.00038
1	Play→No-play	0.00053
2	No-play→Play	0.00092
2	Play→No-play	0.00054
3	No-play→Start Whistle	0.00117
3	Start Whistle→Play	0.04973
3	Play→Stop Whistle	0.00050
3	Stop Whistle→No-play	0.04709

Table 4: Mean transition probabilities for each camera.

optic flow cue, normalized to have zero mean and unit variance, and the softmax confidence for the play state from our deep visual classifier. For Camera 3, four conditional distributions are shown, including the distributions for start and stop whistles, to use in our 4-state HMM. Note the superior discriminative power of the deep visual cue. Fig. 10 shows the conditional densities for the auditory cue (log of the Weiner filter 3 response, normalized to have zero mean and unit variance).

Note that the state transition probabilities and emission distributions used in our HMMs will vary slightly with each fold of our k-fold cross-validation.

We employ the Viterbi algorithm [16] to efficiently determine the maximum a posteriori sequence of hidden states given the observations. One limitation of this approach is that it treats all errors equally, whereas one might expect that mislabeling a play state as a no-play state might be more serious than mislabeling a no-play state as a play state, as the former could lead to the viewer missing a key part of the game, whereas the latter would just waste a little time.

To handle this issue, we employ a play bias parameter $\alpha \geq 1$ that modifies the transition matrix to upweight the probability of transitions to the play state, downweighting other transitions so that each row still sums to 1. Varying this parameter allows us to sweep out a precision-recall curve for each camera. To compress the videos we simply retain any frames estimated to be play frames and excise any frames estimated to be no-play frames.

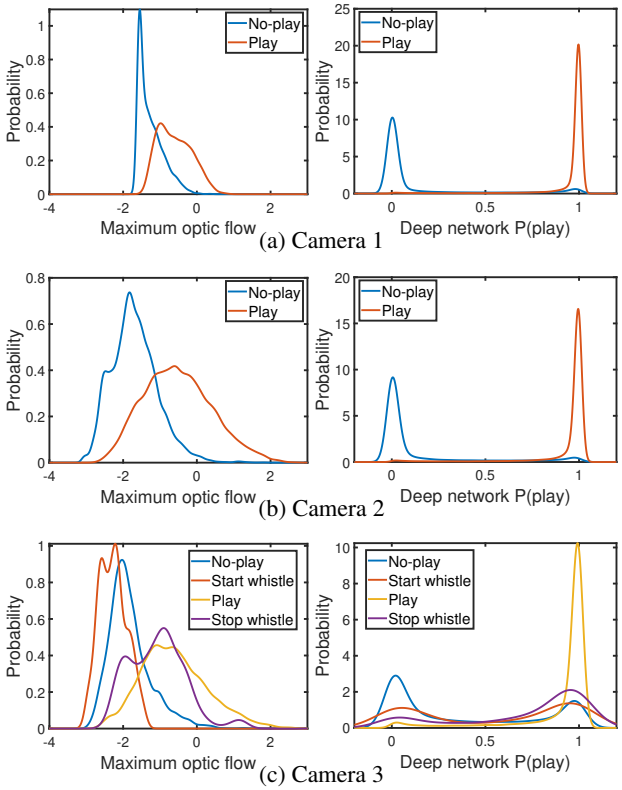


Figure 9: Conditional probability densities for the maximum optic flow (left) and the deep network $P(play)$ (right) visual cues on Game 1 from each of the three cameras..

7. Evaluation

We evaluate our approach using precision-recall for retaining play frames (Cameras 1 and 2) and retaining play and whistle frames (Camera 3):

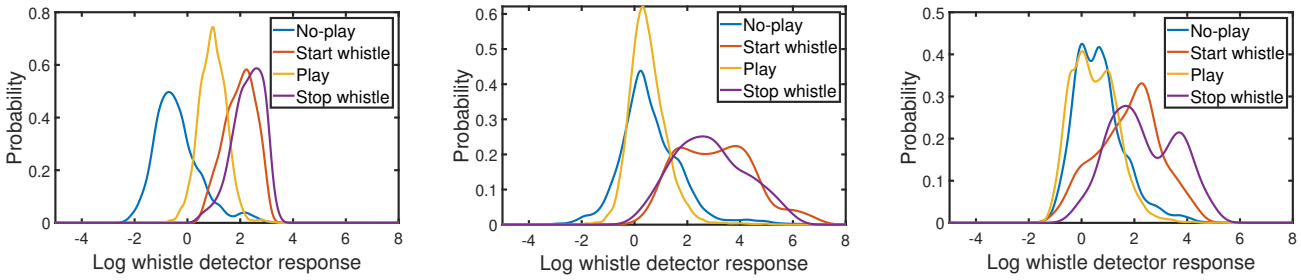
$$\text{Precision} = \frac{\text{\# play \& whistle frames retained}}{\text{\# frames retained}} \quad (6)$$

$$\text{Recall} = \frac{\text{\# play \& whistle frames retained}}{\text{\# play \& whistle frames in video}} \quad (7)$$

We also evaluate the %compression at each rate of recall.

Fig 11 shows results, averaged over all leave-one-game-out folds. For Camera 3, we evaluated using a 2-state HMM with only visual cues as well as a 4-state HMM with both visual and audio cues. For reference we show as a lower bound the performance of a baseline that excises random frames, and as an upper bound the compression-recall attained by an ideal model that first excises all non-play frames before beginning to excise play frames.

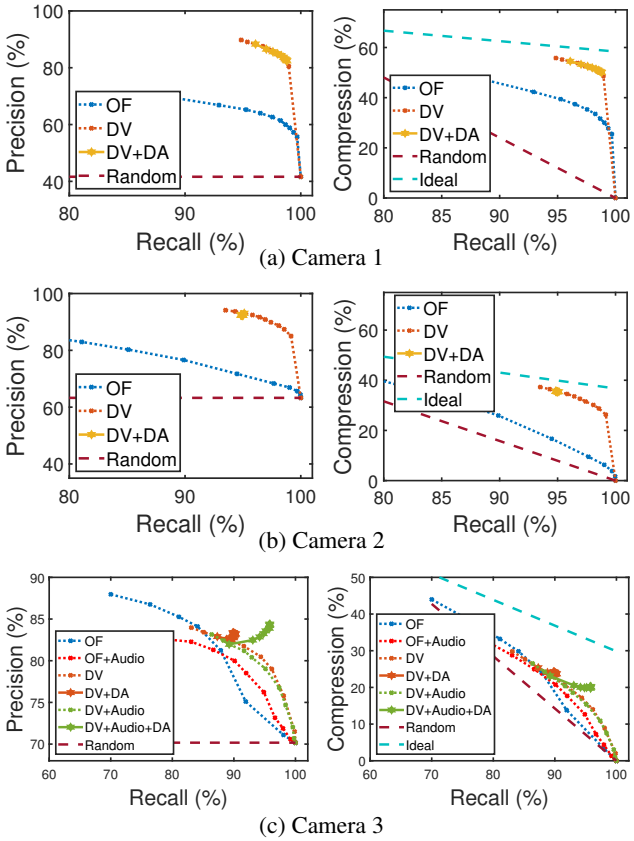
The deep visual cue clearly outperforms the optic flow cue for all cameras. Interestingly, while the the optic flow cue clearly benefits from integration with the audio cue, the deep visual cue seems to be strong enough on its own, and no sensory integration benefit is observed. Fig 12 shows



(a) Game 1 (b) Game 2 (c) Game 3

Figure 10: Conditional densities for Wiener filter 3 whistle detector on games from Camera 3.

that these deep visual cues generalize well across the three camera systems.



(a) Camera 1

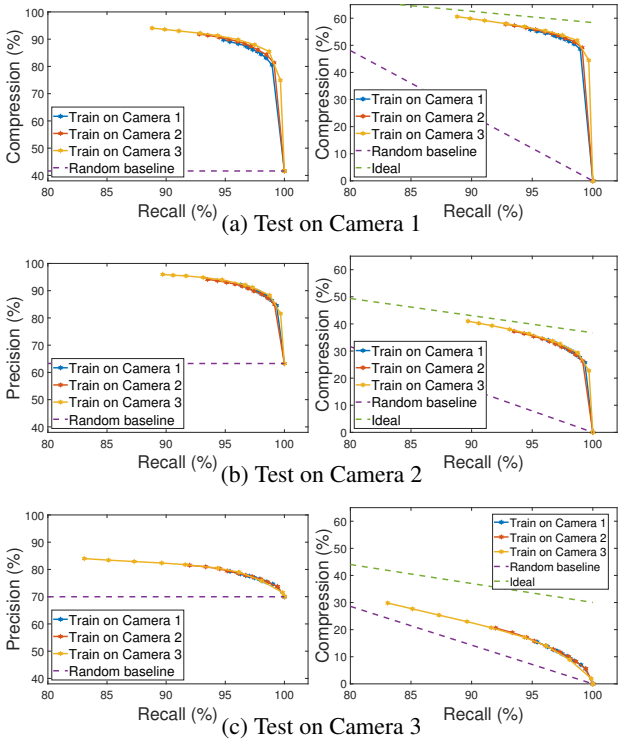
(b) Camera 2

(c) Camera 3

Figure 11: HMM cross-validated performance. OF: Optical flow. DV: Deep visual feature. DA: Domain adaptation. The random baseline assumes excision of random frames, while ideal assumes excision of no-play frames prior to excision of any play frames.

8. Domain Adaptation

We failed to observe any benefit of visuo-auditory integration for Camera 3 once we used strong visual cues. What might explain this? One possibility is domain shift.



(a) Test on Camera 1

(b) Test on Camera 2

(c) Test on Camera 3

Figure 12: Performance of deep visual cues across different camera systems. Left: Precision-recall curves. Right: Compression-recall curves.

While the deep visual cues generalized well across cameras (Fig 12), the auditory emission distributions for Camera 3 are seen to vary substantially across games (Fig 10). This led us to wonder whether we could adapt the HMM at inference to accommodate these domain shifts.

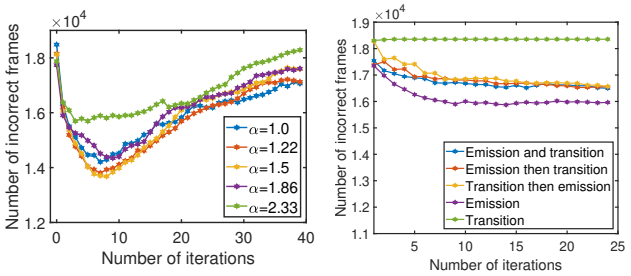
The standard approach to unsupervised HMM parameter learning is the Baum-Welch algorithm [10]. Here we employed the faster and simpler Viterbi training (segmental k-means) method [7], which allowed us to quickly assess different forms of constrained unsupervised adaptation to handle domain shift.

Specifically, for each test game we first use the Viterbi algorithm to compute the MAP solution using the transition

and emission distributions from the training games. We then use the predicted states as pseudo-labels to update the emission and/or the transition distributions using only the test data. We then re-run the Viterbi algorithm and iterate. We explored 5 different adaptation schemes:

1. **Emission & transition** - Simultaneous update of both emission and transition distributions on every iteration.
2. **Emission then transition** - Update emission and then transition distributions on alternate iterations.
3. **Transition then emission** - Update transition and then emission distributions on alternate iterations.
4. **Emission** - Update only the emission distributions. The transition distributions remain fixed.
5. **Transition** - Update only the transition distributions. The emission distributions remain fixed.

We evaluated for 5 different settings of the bias parameter ($\alpha = 1.0, 1.22, 1.5, 1.86, 2.33$). Since this algorithm is not guaranteed to converge, we ran for 40 iterations, assessing the number of incorrectly labelled frames after each iteration. Fig. 13 shows how performance on Camera 3, as measured by the number of incorrectly labelled frames, varies during adaptation, using the Emission adaptation scheme. For audio-visual adaptation, we see an initial decrease in error, but overfitting ultimately leads to a subsequent increase. For our vision-only system, convergence appears to be monotonic.



(a) (b)

Figure 13: (a) Emission adapation for Camera 3 using both deep visual and auditory cues for a range of α values. (b) Five different forms of adaptation, using only deep visual cues, with $\alpha = 1.5$.

To assess the relative merit of our 5 candidate adaptation schemes, we examined the best performance obtained for audio-visual adaptation, for each scheme and for each α setting over the adaptation process. The results are shown in Table 5. Best performance was obtained with emission-only adaptation and a bias of $\alpha = 1.5$, after 9 iterations.

Fig 11c shows that while both auditory-visual (4-state HMM) and visual-only (2-state HMM) adaptation improve HMM performance for Camera 3, the improvement is most dramatic through the auditory cues, presumably because of the greater domain shift in the auditory distributions.

	Play bias (α) values				
	1.0	1.22	1.5	1.86	2.33
Initial error	18,482	18,152	18,121	17,732	17,874
Emission & transition	14,669	14,337	13,856	14,602	15,745
Emission then transition	14,494	14,401	13,860	15,212	15,853
Transition then emission	14,530	14,738	14,613	14,825	15,764
Emission	14,209	13,803	**13,678**	14,345	15,692
Transition	18,482	18,152	18,121	17,732	17,874

Table 5: Number of incorrect frame classifications averaged across all games recorded using Camera 3, for each adaptation method applied to audio-visual cues, for selected values of play bias α.

We also evaluated visual domain adaptation on Cameras 1 and 2, but found little improvement (Fig 11a,11b), perhaps because the conditional densities for the deep visual cue were already very well separated (Fig 9a, 9b) and thus robust to visual domain shift between cameras.

9. Run time

Optic flow runs at 95 fps and the faster-RCNN detector runs at 24 fps on a 1080Ti GPU and a 4.2GHz i7-7700K CPU. The HMM runs at 600 fps on the CPU.

10. Conclusions & Future Work

We have developed a novel system for automatic play-break segmentation for hockey, and have shown its utility in abbreviating hockey videos while maintaining high recall for periods of active play. We found that with a modest dataset it is possible to train a small visual deep network to produce visual cues for play/no-play classification that are much more reliable than a simple optic flow cue. Incorporation of an HMM framework accommodates statistical dependencies over time, allowing effective play/break segmentation and temporal video compression. We found that integration of auditory (whistle) cues could boost segmentation performance, but only by incorporating unsupervised adaptation of emission distribution models to accommodate domain shift. Our system was found to achieve temporal compression rates of 20-50% at a recall of 96%.

Future work will compare Viterbi training to Baum-Welch updates for domain adaptation, to assess performance-speed tradeoffs.

11. Acknowledgements

We acknowledge the support of NSERC CREATE DAV (www.createdav.ca), the York VISTA (vista.info.yorku.ca/) and Research Chair programs. We thank Canlan Ice Sports (www.canlansports.com) and York University Athletics for providing the video data for this project.

References

[1] M. A. Carbonneau, A. J. Raymond, E. Granger, and G. Gagnon. Real-time visual play-break detection in sport events using a context descriptor. In *IEEE International Symposium on Circuits and Systems (ISCAS)*, pages 2808–2811. IEEE, 2015.

[2] J. Chen and P. Carr. Mimicking human camera operators. In *IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 215–222. IEEE, 2015.

[3] J. Chen, H. M. Le, P. Carr, Y. Yue, and J. J. Little. Learning online smooth predictors for real-time camera planning using recurrent decision trees. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4688–4696. IEEE, 2016.

[4] A. Ekin and M. Tekalp. Generic play-break event detection for summarization and hierarchical sports video analysis. In *International Conference on Multimedia and Expo (ICME)*, volume 1, pages I–169. IEEE, 2003.

[5] G. Farnebäck. Two-frame motion estimation based on polynomial expansion. In *Scandinavian Conference on Image Analysis (SCIA)*, pages 363–370. Springer, 2003.

[6] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *IEEE conference on Computer Vision and Pattern Recognition (CVPR)*, pages 770–778, 2016.

[7] B.H. Juang and L.R. Rabiner. The segmental k-means algorithm for estimating parameters of hidden Markov models. *IEEE Transactions on Acoustics, Speech and Signal Processing*, 38(9):1639–1641, 1990.

[8] S. M. Kay. *Fundamentals of statistical signal processing: detection theory*. Prentice Hall PTR, 1998.

[9] H. Pidaparthy and J. H. Elder. Keep your eye on the puck: automatic hockey videography. In *IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 1636–1644. IEEE, 2019.

[10] L. R. Rabiner. A tutorial on hidden Markov models and selected applications in speech recognition. *Proceedings of the IEEE*, 77(2):257–286, 1989.

[11] S. Ren, K. He, R. Girshick, and J. Sun. Faster R-CNN: Towards real-time object detection with region proposal networks. *arXiv preprint arXiv:1506.01497*, 2015.

[12] B. W. Silverman. *Density estimation for statistics and data analysis*, volume 26. CRC press, 1986.

[13] M. Tavassolipour, M. Karimian, and S. Kasaei. Event detection and summarization in soccer videos using Bayesian network. *IEEE Transactions on Circuits and Systems for Video Technology*, 24(2):291–304, 2013.

[14] D. Tjondronegoro, Y. P. Chen, and B. Pham. The power of play-break for automatic detection and browsing of self-consumable sport video highlights. In *ACM SIGMM International Workshop on Multimedia Information Retrieval*, pages 267–274.

[15] D. Tjondronegoro, Y. P. Chen, and B. Pham. Sports video summarization using highlights and play-breaks. In *ACM SIGMM International Workshop on Multimedia Information Retrieval*, pages 201–208, 2003.

[16] A. Viterbi. Error bounds for convolutional codes and an asymptotically optimum decoding algorithm. *IEEE Transactions on Information Theory*, 13(2):260–269, 1967.

[17] L. Xie, P. Xu, S. Chang, A. Divakaran, and H. Sun. Structure analysis of soccer video with domain knowledge and hidden markov models. *Pattern Recognition Letters*, 25(7):767–775, 2004.

[18] Z. Zhang. A flexible new technique for camera calibration. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 22(11):1330–1334, 2000.