

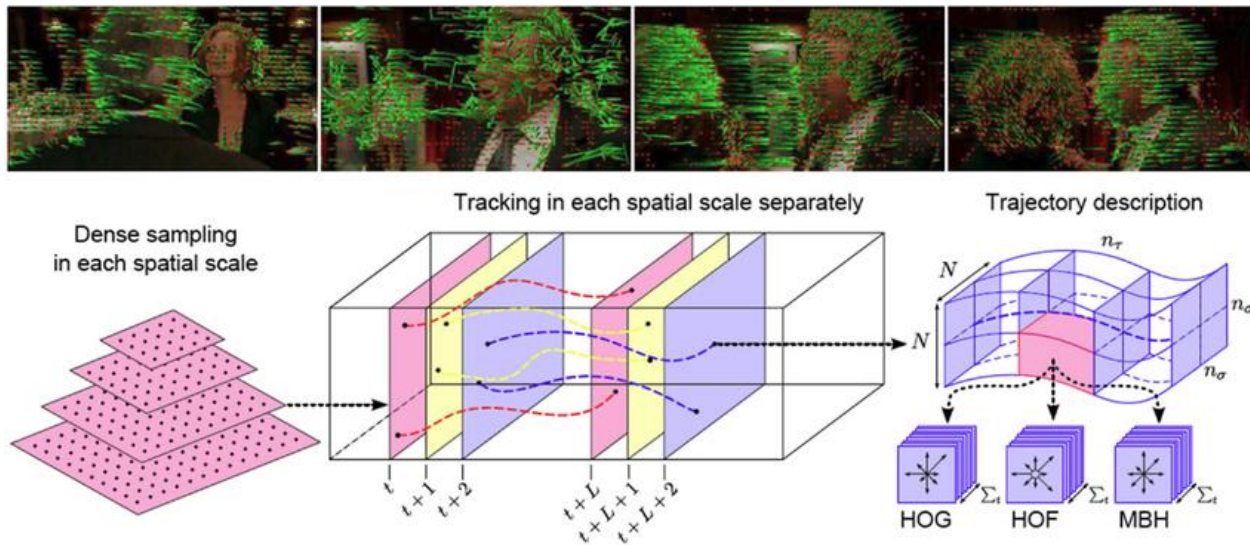
COMPRESSED-DOMAIN VIDEO CLASSIFICATION WITH DEEP NEURAL NETWORKS: “THERE’S WAY TOO MUCH INFORMATION TO DECODE THE MATRIX”

Aaron Chadha, Alhabib Abbas, Yiannis Andreopoulos



Background: Action recognition

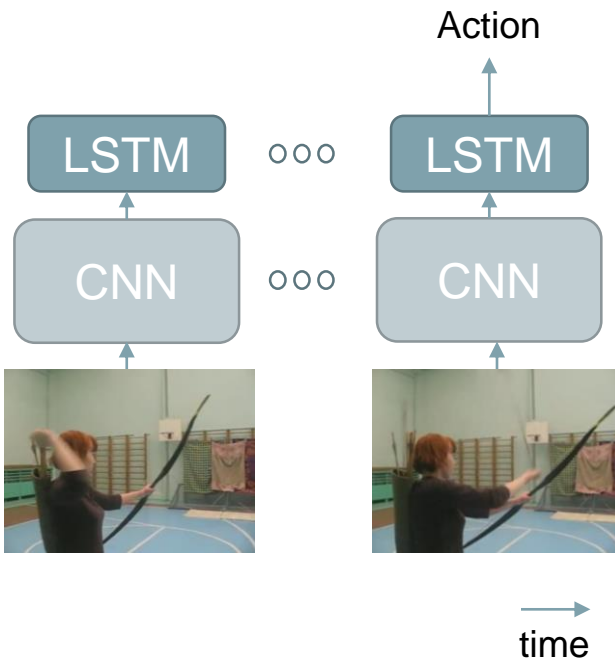
- Action recognition: Image sequences \rightarrow Actions
- Before deep learning – dense trajectories using optical flow:



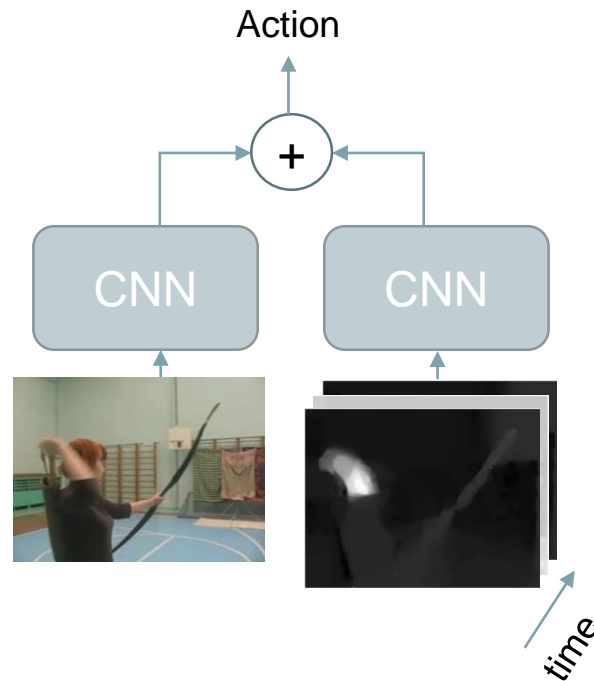
Background: Action Recognition

State-of-the-art deep learning methods:

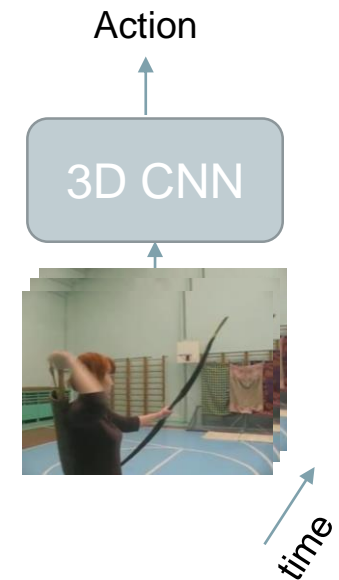
a) LSTM + CNN



b) Flow + RGB



c) 3D RGB CNN



Common Issues

- RGB frame inputs require full video decoding
- Optical flow is expensive to compute (per-pixel)
- LSTMs are notoriously slow to train
- 3D CNN on (large) RGB frames requires heavy processing
- Short temporal extent of inputs -> only looking at local motion cues and not long term dependencies
- Redundancy between consecutive frames

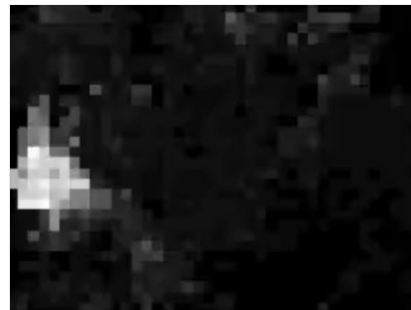
Our Proposal

- We would like to use the video codec directly as a *spatio-temporal sensor*
- We propose a 3D convolutional neural network that directly ingests motion vector flow extracted from the encoder bitstream

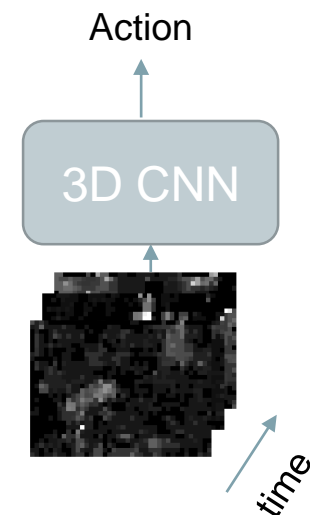
Video



MV flow



Proposal



3D-CNN Input

- Macroblock (MB) bitstream representation from codec:

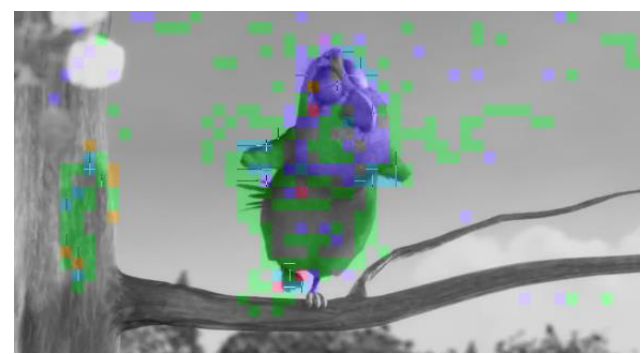
| ADDR | TYPE | QUANT | VECTOR | CBP | b0 | b1 | ... b5 |

- Addr: Block address in image
- Type: Intra(I), inter (P), or bi-directional inter (B) frames
- Vector: Motion vector

```

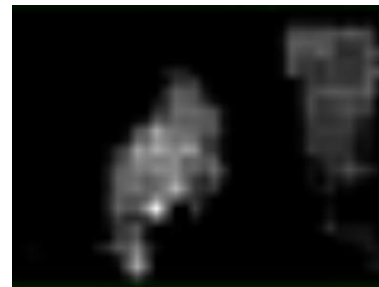
00000000 00 00 00 20 66 74 79 70 69 73 6F 6D 00 00 02 00 69 73 6F 6D 69 73 6F 32 61 76 63 31 6D 70 34 31 ... ftypisom...isomiso2avc1mp41
00000020 00 00 00 08 66 72 65 65 00 98 A5 0D 6D 64 61 74 00 00 02 AB D6 05 FF FF AA DC 45 E9 BD B6 D9 48 ....free...mdat.....E...H
00000040 B7 96 2C D8 20 D9 23 EE EF 78 32 36 34 20 2D 20 63 6F 72 65 20 31 34 38 20 72 32 37 34 34 20 62 ...._...x264 - core 148 r2744 b
00000060 39 37 61 65 30 36 20 2D 20 48 2E 32 36 34 2F 4D 50 45 47 2D 34 20 41 56 43 20 63 6F 64 65 63 20 07ae06 - H.264/MPEG-4 AVC codec
00000080 2D 20 43 6F 70 79 6C 65 66 74 20 32 30 33 2D 32 30 31 36 20 2D 20 68 74 74 70 3A 2F 2F 77 77 - Copyright 2003-2016 - http://w
00000100 77 2E 76 69 64 65 6F 6E 61 6E 2E 6F 72 67 2F 82 36 24 2E 68 74 6D 6C 20 2D 20 6F 70 74 69 6E www.videolan.org/v264.html - opio
00000120 6E 73 3A 20 63 61 62 61 63 3D 31 20 72 65 66 3D 33 20 64 65 62 6C 6F 63 68 3D 31 3A 30 3A 20 ns: cabac=1 ref=3 deblock=1:0:0
00000140 61 6E 61 6C 79 73 65 3D 30 78 33 3A 30 78 31 31 33 20 6D 65 3D 68 65 78 20 73 75 62 6D 65 3D 31 analyse=0x3:0x113 me=hex subme=7
00000160 20 70 73 79 3D 31 20 70 73 79 5F 72 64 3D 31 2E 30 30 3A 30 2E 30 30 2D 6D 69 78 65 64 5F 72 65 psy=1 psy_rd=1.00:0.00 mixed_re
00000180 66 3D 31 20 6D 65 5F 72 61 6E 67 65 3D 31 36 20 63 68 72 6F 6D 61 5F 6D 65 3D 31 20 74 72 65 6C f=1 me_range=16 chroma_me=1 trel
00000200 6C 69 73 3D 31 20 38 78 38 64 63 74 3D 31 20 63 71 6D 3D 30 20 64 65 61 64 7A 6F 6E 65 3D 32 31 lis=1 8x8dct=1 cqm=0 deadzone=21
00000220 2C 31 31 20 66 61 73 74 5F 70 73 68 69 70 3D 31 20 63 68 72 6F 6D 61 5F 71 70 5F 6F 66 66 73 65 !1 fast_pskip=1 chroma_qp_offse
00000240 74 3D 2D 32 20 74 68 72 65 61 64 73 3D 36 20 6C 6F 6F 6B 61 68 65 61 64 5F 74 68 72 65 61 64 73 --2 threads=6 lookahead_threads
00000260 3D 31 20 73 6C 69 63 65 64 5F 74 68 72 65 61 64 73 3D 30 20 6E 72 3D 30 20 64 65 63 69 6D 61 74 -1 sliced_threads=0 nr=0 decimat
00000280 65 3D 31 20 69 6E 74 65 72 6C 61 63 65 64 3D 30 20 62 6C 75 72 61 79 3F 63 6F 6D 70 61 74 3D 30 e=1 interlaced=0 bluray_compat=0
00000300 20 63 6E 6E 73 74 72 61 69 6E 65 64 5F 69 6E 74 72 61 3D 30 20 62 66 72 61 6D 65 73 3D 33 20 6E constrained_intra=0 bframes=3 b
00000320 5F 70 75 72 61 6D 69 64 3D 32 20 62 5F 61 68 61 70 74 3D 31 20 62 5F 62 69 61 73 3D 30 20 64 69 pyramid=2 b_adapt=1 b_bias=0 di
00000340 72 65 63 74 3D 31 20 77 65 69 67 68 74 62 3D 31 20 6F 70 65 6E 5F 67 6F 70 3D 30 20 77 65 69 61 rect=1 weightb=1 open_gop=0 weig
00000360 68 74 70 3D 32 20 68 65 79 69 6E 74 3D 32 35 30 20 6B 65 79 69 6E 74 5F 6D 69 6E 3D 32 34 20 73 htp=2 keyint=250 keyint_min=24 s
00000380 63 65 6E 65 63 75 74 3D 34 30 20 69 6E 74 72 61 5F 72 65 66 72 65 73 68 3D 30 20 72 63 5F 6C 6E cencut=40 intra_refresh=0 rc_lo
00000400 6F 6B 61 68 65 61 64 3D 34 30 20 72 63 3D 63 72 6E 20 6D 62 74 72 65 65 3D 31 20 63 72 66 3D 32 bkahead=40 rc=crf mbtree=1 crf=2
00000420 33 2E 30 71 63 6F 6D 70 3D 30 20 71 70 6D 69 6E 3D 30 20 71 70 6D 61 78 3D 36 39 20 3.0 qcqcomp=0.60 qpmin=0 qpmax=69
00000440 71 70 73 74 65 70 3D 34 20 69 70 5F 72 61 64 69 6F 3D 31 2E 34 30 20 61 71 3D 31 3A 31 2E 30 30 upstep4 ip_ratio=1.40 aq=1:1.00
00000460 00 80 00 2A AC 65 88 84 00 5F FE A5 C5 C0 7B 17 50 1C 57 FF FF D9 F1 07 05 F5 F9 F7 BB 84 B4 .....e.....{P.W.....
00000480 5E 2E BC 2A EA 8D 99 6E 51 C1 20 E7 B2 05 73 2C 00 15 ED 41 4F 84 95 FC AC 76 56 FF 0D 96 6F C4 *...n0...s...A0...v...o.
00000500 5C 4E 54 F4 D4 6A 96 AF 8D 95 2C FD F4 3F 6B 0E 4B 9F ED 9E 9A D6 29 9A 9E E7 03 74 F1 C1 44 FA Wtr: j.....7k.K.....)r..D.
00000520 8F 22 C5 0C B8 76 78 F7 ED 96 67 7C AA 92 47 EC 0D C1 EE DC DB CF EA 55 34 45 D2 13 30 F4 B5 "...V.....|..G.....S.E..0..
00000540 5A 72 10 F4 8D 40 2D 14 5A 97 67 71 91 15 0D 35 4F FC F7 5B FB B4 F2 40 75 36 4D 7C 22 33 85 78 Zl:..8-.z.g.....50..W...8uM13.x
00000560 F6 3B 37 6A 64 5C 02 F3 F6 3C CF 45 8A 2B 9E B2 F9 60 6E 4A E5 2D B5 7C E1 4A 76 D0 6C FB 71 83 ;7jd\...<.E.+...nd.-|.Jv.l.q.
00000580 B1 D0 05 92 15 CD 9D 17 AF 33 9B 68 79 F7 58 F3 52 D3 BE D4 E4 F6 87 85 3E 69 7E B0 33 65 4C 15 .....3.hy.X.R.....>i<=3E.
00000600 06 7F 1A 48 6F 8C B5 7A F8 DB 48 D5 CE BB 9B A7 D5 4D 39 49 63 62 92 FF D8 FC 5C 76 5A CF 35 ...Ho.z..H.....M9Ic.c.....\v2.5
00000620 FF B6 1F 35 51 EB 1A E7 9D AA D7 E2 D4 11 4A 8C C5 20 F1 AC A4 0E CF 11 D0 00 38 3C FD A5 8F 38 ...5Q.....J.....8<...8
00000640 2A 6D BC 1A 0E 59 03 62 2E D3 A4 B8 48 4F F7 CD 86 32 A0 5F 3B 01 03 F5 AB C0 BB C0 C8 EE A3 *m...Y.b....HO..2.2_.....

```



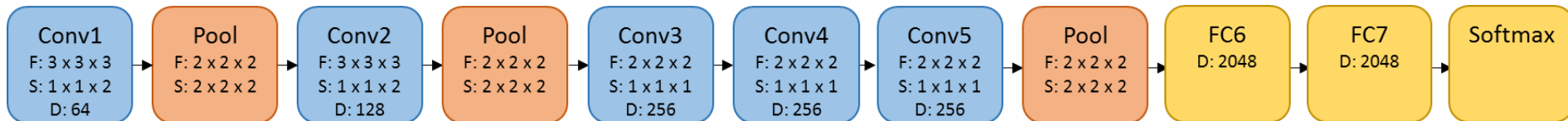
3D-CNN Input

- Only ingest P-frames
- 8 x 8 macroblock size (equates to 40 x 30 MV frame representations on UCF-101)
- 2 channels, δx and δy
- Low spatial resolution -> longer temporal extent
- CNN input is 4D: 24 x 24 x 2 x 160 resolution



3D-CNN Architecture

- 3D CNN (F = filter, S = stride, D = depth):



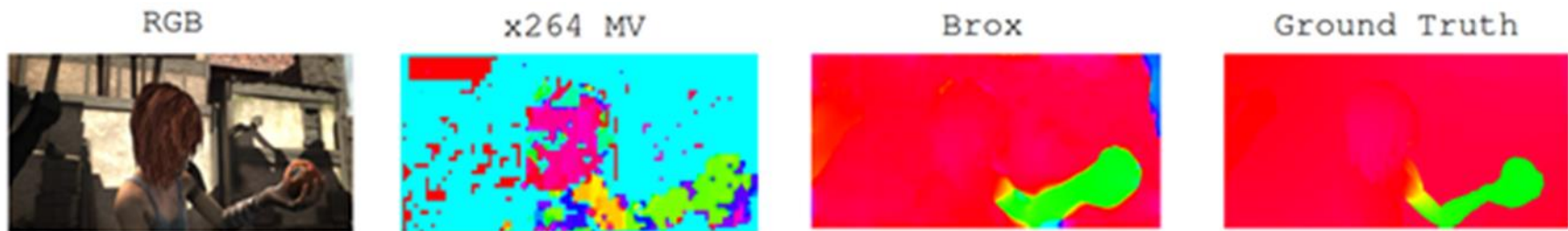
Temporal stride of 2 to quickly downsample input

2 x 2 x 2 conv filters to minimize weights, sufficiently large to cover spatial extent

3 x 3 x 5 input to fully connected layers

Experimental Evaluation

- Visual comparison of inputs (MPI-Sintel):



- Visual quality measured in terms of EPE (lower is better)

Input	Runtime per frame (ms)		% P	% NZ	EPE
	Decoding	Flow Estimation			
Proposed	0	0.16 (CPU)	62	21	15.26
Brox	3.08 (CPU)	6270 (GPU)	–	–	6.32
FlowNet2	3.08 (CPU)	123 (GPU)	–	–	3.14

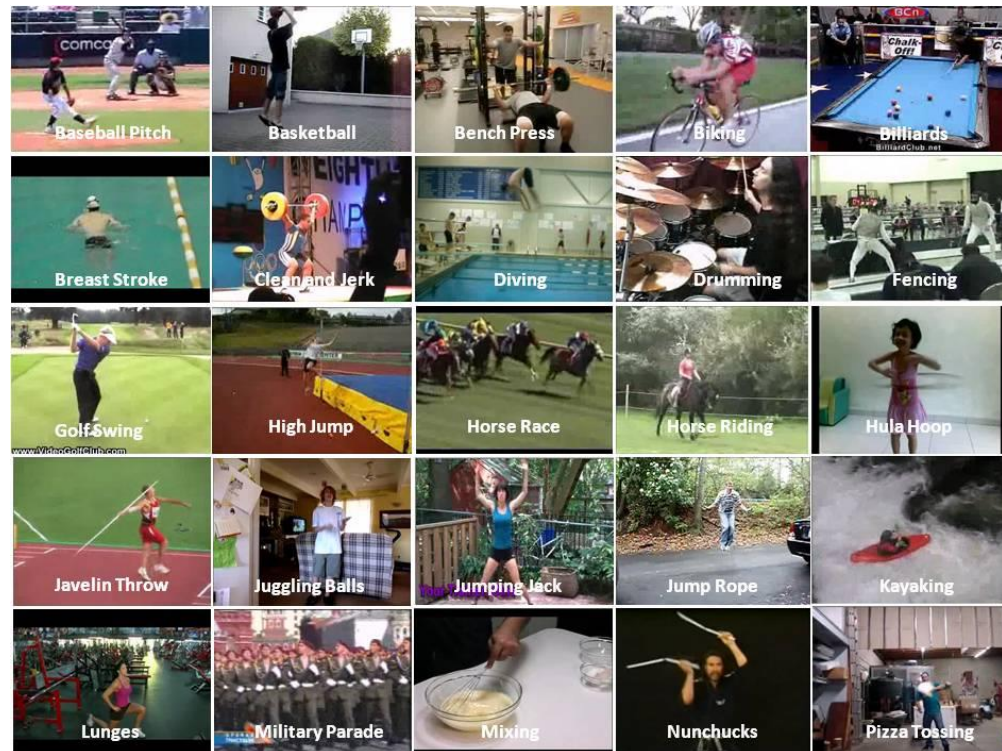
- Proposed MV flow is over 4 times faster than Brox to compute

Experimental Evaluation

- We measure classification performance on UCF-101 and HMDB-51 datasets:

UCF-101: 13k videos,
101 classes,
320 x 240 resolution,
25 FPS

HMDB-51: 7k videos,
51 classes,
320 x 240 resolution,
30 FPS



Experimental Evaluation

- Accuracy compared to state-of-the-art:

Framework	Input Size	Complexity #A, #W ($\times 10^6$)	Accuracy (%)	
			UCF	HMDB
Proposed	$24^2 \times 2 \times 160$	4.0, 29.4	77.5	49.5
SSCNN-Brox	$224^2 \times 20$	2.0, 90.6	83.7	54.6
SSCNN+	$224^2 \times 3$	2.0, 90.6	73.0	40.5
LTC-Brox	$58^2 \times 2 \times 100$	42.1, 12.2	82.6	56.7
LTC-Mpegflow	$58^2 \times 2 \times 60$	25.3, 10.6	63.8*	–
SFCNN+	$170^2 \times 3 \times 10$	1.80, 26.7	65.4	–
C3D+	$112^2 \times 3 \times 16$	30.2, 63.7	82.3	–

Ours:

- 3105 FPS
- 29.4M weights

Best acc.:

- 185 FPS
- 90.6M weights

Conclusions

- MV flow extraction (P-frames only) is up to 4 orders of magnitude faster than optical flow variants
- Low spatial resolution is counter-balanced by very long temporal extent (160 frames)
- We achieve competitive accuracy (77.5% on UCF-101) to methods using optical flow
- Lightweight 3D CNN = up to an order of magnitude faster processing than recent work
- Code available at:

<https://github.com/mvcnn>

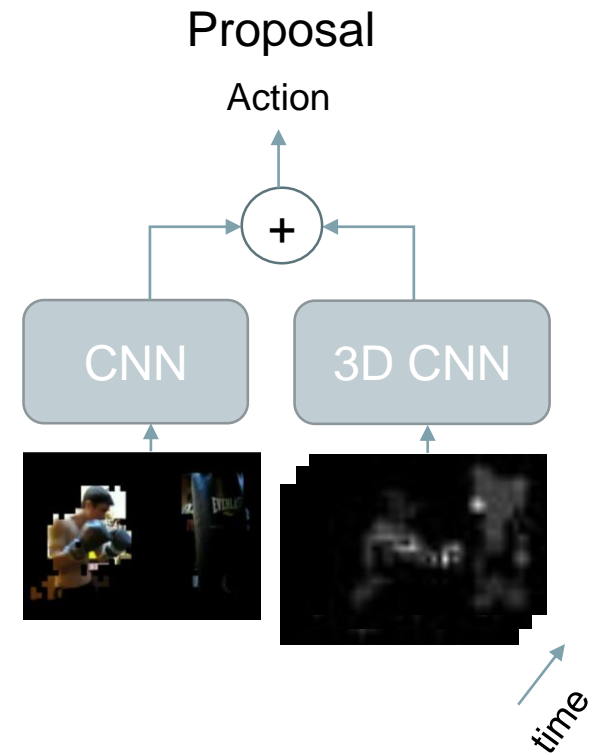
Further work

- We have since extended to a two-stream architecture using selectively decoded RGB frames:

Video



Selective decoding



Further work

Framework	Accuracy (%)	
	UCF	HMDB
Proposed, $X = 10$	89.8	56.0
Proposed, $X = 50$	88.9	54.6
TSCNN (avg. fusion)	86.9	58.0
TSCNN (SVM fusion)	88.0	59.4
CNN-pool	88.2	–
C3D (3 nets)+IDT	90.4	–
LTC	91.7	64.8
EMV + RGB-CNN	86.4	–
IP+SVM	–	59.5
Line Pooling	88.9	62.2

End-to-end cost
(ours):
\$0.228 – \$0.250

End-to-end cost
(Zisserman et al.):
\$11.103

Note: Cost is reported on evaluation for UCF-101, using AWS p2.xlarge and r3.xlarge instance prices as appropriate