Recurrent Space-time Graph Neural Networks

Andrei Nicolicioiu* [1], Iulia Duta* [1], Marius Leordeanu[1 2 3]

[1]Bitdefender, [2]University Politehnica of Bucharest, [3]Institute of Mathematics of the Romanian Academy

Contributions

- propose a **general neural graph** block for learning in spatio-temporal domain, used as an intermediary block within any model.
- **factorize space and time** and process them differently from an unstructured video, achieving **low computational complexity**
- introduce a synthetic dataset involving **explicit space-time interactions**: SyncMNIST
- **state-of-the-art** results on real world dataset, **Something-Something**

Our approach:

- neural graph **recurrent in space and time**
- extract features from **fixed regions** in video and use them as **nodes** in a **graph** model
- process video by message passing to get **long range interactions**: Space and Time Stages

Algorithm

Algorithm 1 Space-time processing in RSTG

Input: Features $F \in R^{T \times H \times W \times C}$

repeat

$\mathbf{v}_i \leftarrow \textit{extract\_features}(F_t, i) \qquad \forall i$

for $k = 0$ **to** $K-1$ **do**

$\mathbf{v}_i = \mathbf{h}_i^{t,k} = \mathrm{f}_{\mathrm{time}}(\mathbf{v}_i, \mathbf{h}_i^{t-1,k}) \qquad \forall i$

$\mathbf{m}_{j,i} = \mathrm{f}_{\mathrm{send}}(\mathbf{v}_j, \mathbf{v}_i) \qquad \forall i, \forall j \in \mathcal{N}(i)$

$\mathbf{g}_i = \mathrm{f}_{\mathrm{gather}}(\mathbf{v}_i, \{\mathbf{m}_{j,i}\}_{j \in \mathcal{N}(i)}) \qquad \forall i$

$\mathbf{v}_i = \mathrm{f}_{\mathrm{space}}(\mathbf{v}_i, \mathbf{g}_i) \qquad \forall i$

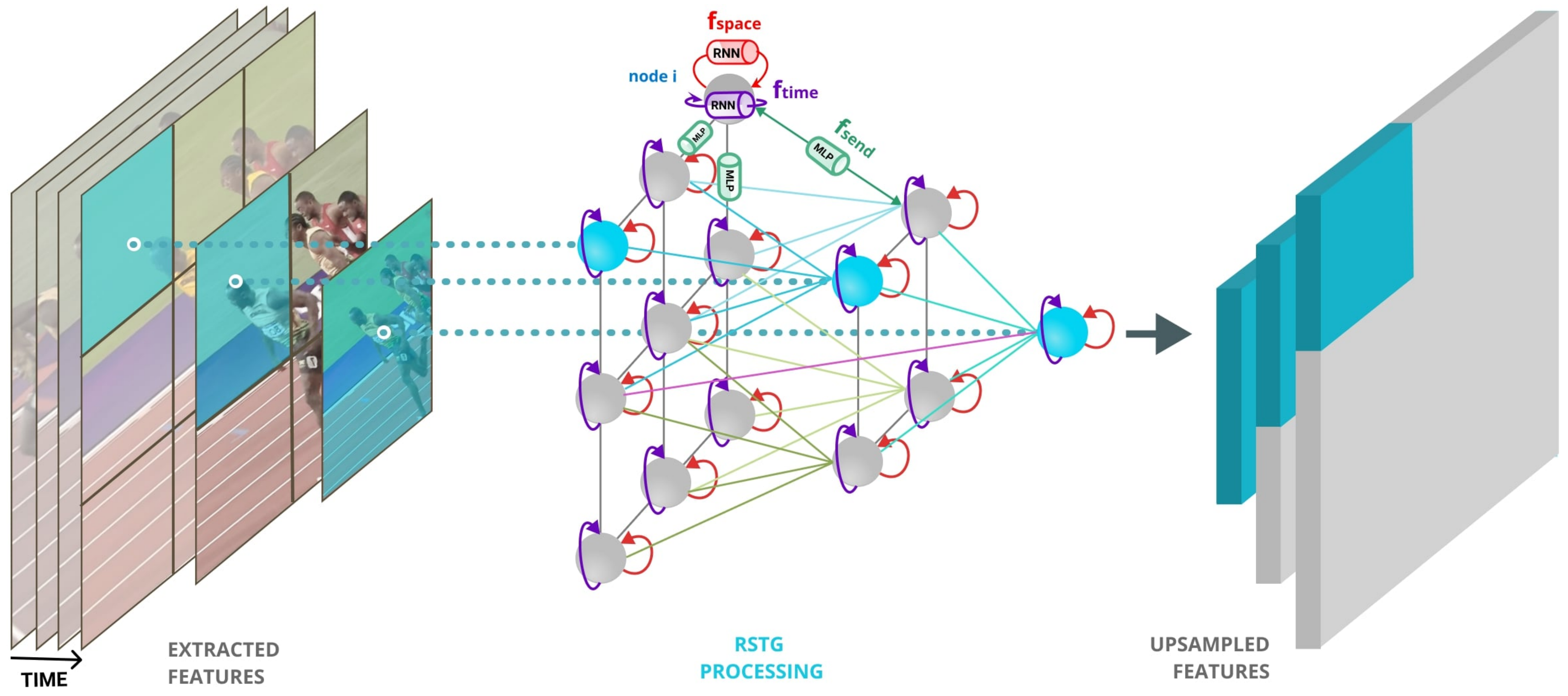
end for

$\mathbf{h}_i^{t,K} = \mathrm{f}_{\mathrm{time}}(\mathbf{v}_i, \mathbf{h}_i^{t-1,K}) \qquad \forall i$

$t = t + 1$

until end-of-video

$\mathbf{v}_{final} = f_{aggregate}(\{\mathbf{h}_i^{1:T,K}\}_{\forall i})$

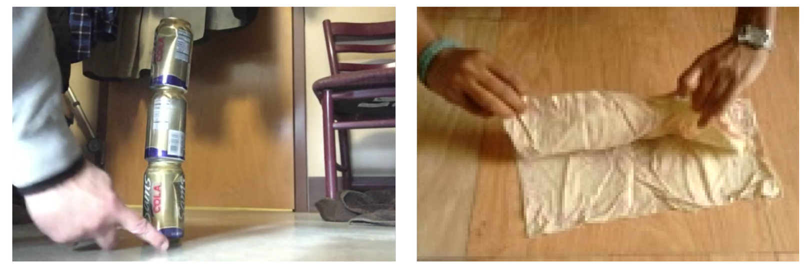


Recurrent factorized Graph Nets are suited for video analysis tasks heavily relying on interactions.

Accuracy on Smt-Smt-v1 dataset

Model	Top-1	Top-5
C2D	31.7	64.7
TRN [1]	34.4	-
RSTG - C2D	**42.8**	**73.6**
MFNet-C50 [2]	40.3	70.9
I3D [3]	41.6	72.2
NL I3D [3]	44.4	76.0
NL I3D + Joint GCN [3]	46.1	76.8
$\mathrm{ECO}_{Lite-16F}$ [4]	42.2	-
MFNet-C101 [2]	43.9	73.1
I3D [5]	45.8	76.5
S3D-G [5]	48.2	78.7
RSTG-to-vec	47.7	77.9
RSTG-to-map res2	46.9	76.8
RSTG-to-map res3	47.7	77.8
RSTG-to-map res4	48.4	78.1
RSTG-to-map res3-4	**49.2**	**78.8**

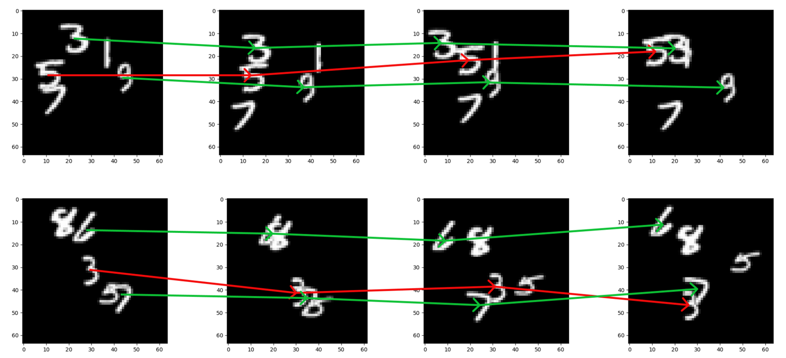
Something-Something is a real world dataset involving human-object interactions.



Accuracy on SyncMNIST datasets

Model	3Sync	5Sync
Mean + LSTM	77.0	-
Conv + LSTM	95.0	39.7
I3D [6]	-	90.6
Non-Local [7]	-	93.5
RSTG: Space-Only	61.3	-
RSTG: Time-Only	89.7	-
RSTG: Homogenous	95.7	58.3
RSTG: 1-temp-stage	97.0	74.1
RSTG: All-temp-stages	**98.9**	94.5
RSTG: Positional All-temp	-	**97.2**

We designed a synthetic dataset where the complexity comes from the necessity of explicitly modeling spatial and temporal interactions but in a clear, simple environment. The goal is to detect a pair of digits that move synchronous.



Graph Creation

- extract features from 2D / 3D **backbone**
- arrange **regions in grids** at multiple scales
- each node receives information **pooled** from a region
- the nodes are **connected** if they come from neighbouring or overlapping regions

Space Processing Stage

Consists of three phases, **recurrently** applied at **each time step**:

- Send: messages represent pairwise spatial interactions

$$\mathrm{f}_{\mathrm{send}}(\mathbf{v}_j, \mathbf{v}_i) = \mathrm{MLP}_s([\mathbf{v}_j|\mathbf{v}_i])$$

- Gather: aggregate received messages by an attention mechanism

$$\mathrm{f}_{\mathrm{gather}}(\mathbf{v}_i) = \sum_{j \in \mathcal{N}(i)} \alpha(\mathbf{v}_j, \mathbf{v}_i) \mathrm{f}_{\mathrm{send}}(\mathbf{v}_j, \mathbf{v}_i)$$

- Update: incorporate global context into each local information

$$\mathrm{f}_{\mathrm{space}}(\mathbf{v}_i) = \mathrm{MLP}_u([\mathbf{v}_i|\mathrm{f}_{\mathrm{gather}}(\mathbf{v}_i)])$$

Positional Awareness:

- each source node should be aware of the destination node's position
- we concatenate the position of both nodes to the input of $\mathrm{f}_{\mathrm{send}}$
- position is represented by a gaussian heatmap centered in node's location

Time Processing Stage

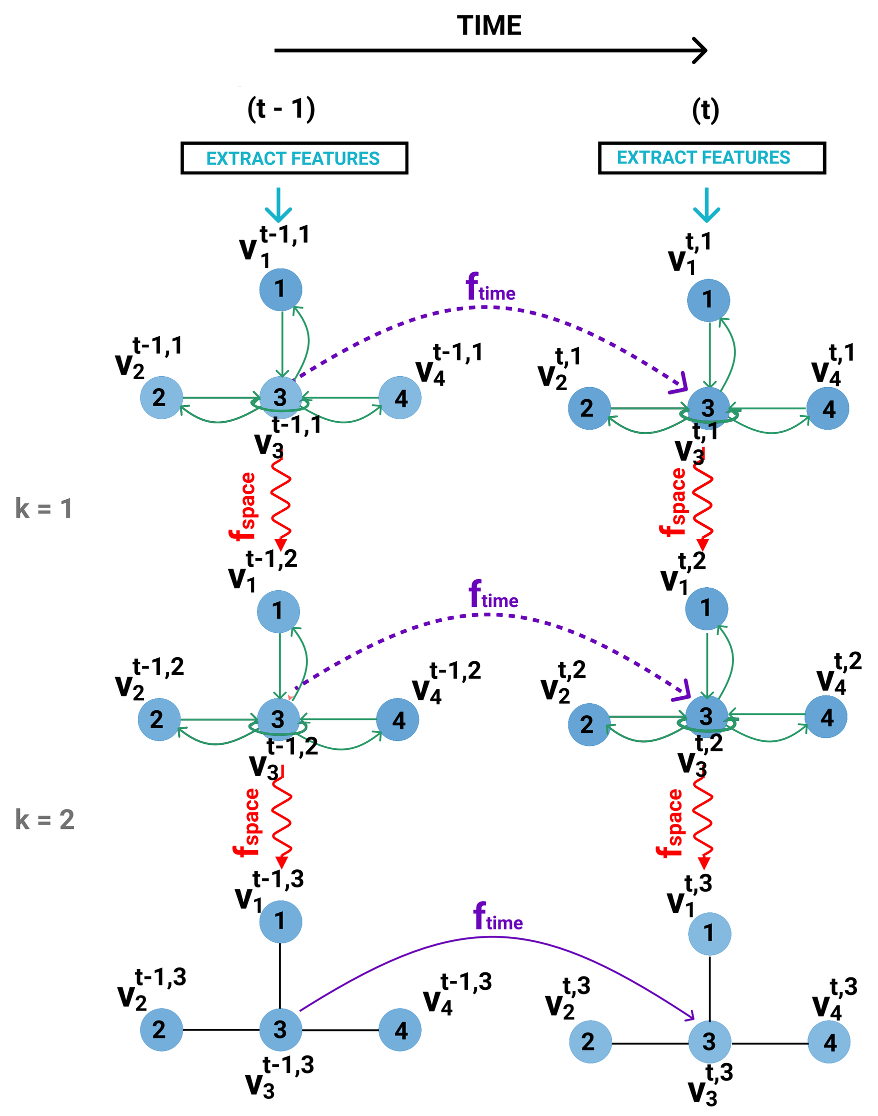
- **across time**, each node incorporates current spatial info into the previous time step features
- each node updates its spatial information using a recurrent function
- no messages exchanged between different regions

$$\underbrace{\mathbf{h}_i^{t,k}}_{time} = \mathrm{f}_{\mathrm{time}}(\underbrace{\mathbf{v}_i^{k}}_{space}, \underbrace{\mathbf{h}_i^{t-1,k}}_{time})$$

Versatile Usage

RSTG model can be used in two ways:

- **RSTG-to-vec**: obtain **1D vector** by summing the all the nodes from the last time step
- **RSTG-to-map**: obtain **features volume** with the same size as the input, by projecting back each node into initial corresponding region
  - more **flexible** model, by incorporating it as a module inside any other architecture



- go from **local to more global** processing by recurrently having multiple space iterations
- **more expressive** power is obtained by alternating Time Processing Stages with Space Processing Stages
- use K alternating stages + a final time stage

References

[1] Zhou et al. ECCV 2018,
[2] Lee et al. ECCV 2018,
[3] Wang and Gupta ECCV 2018,
[4] Zolfaghari et al. ECCV 2018,
[5] Xie et al. ECCV 2018,
[6] Carreira and Zisserman CVPR 2017,
[7] Wang et al. CVPR 2018