



(51) International Patent Classification:

G06K 9/00 (2006.01)

(21) International Application Number:

PCT/EP2019/072348

(22) International Filing Date:

21 August 2019 (21.08.2019)

(25) Filing Language:

English

(26) Publication Language:

English

(71) Applicants: TOYOTA MOTOR EUROPE

[BE/BE]; Avenue du Bourget 60, 1140 BRUSSELS (BE). RHEINISCHE FRIEDRICH-WILHELMS-UNIVERSITÄT BONN [DE/DE]; Regina-Pacis-Weg 3, 53113 BONN (DE).

(72) Inventors: MINCIULLO, Luca; c/o TOYOTA MOTOR EUROPE, Technical Strategic Planning Operation, Avenue du Bourget 60, 1140 BRUSSELS (BE). SOURJ, Yaser; c/o RHEINISCHE FRIEDRICH-WILHELMS-UNIVERSITÄT BONN, Computer Vision Group, Regina-Pacis-Weg 3, 53113 BONN (DE). GALL, Jürgen; c/o RHEINISCHE FRIEDRICH-WILHELMS-UNIVERSITÄT BONN, Computer Vision Group, Regina-Pacis-Weg 3, 53113 BONN (DE). BISWAS, Sovan; c/o RHEINISCHE FRIEDRICH-WIL-

HELMS-UNIVERSITÄT BONN, Computer Vision Group, Regina-Pacis-Weg 3, 53113 BONN (DE).

(74) Agent: DELUMEAU, François et al.; CABINET BEAU DE LOMENIE, 158 Rue de l'Université, 75340 PARIS CEDEX 07 (FR).

(81) Designated States (unless otherwise indicated, for every kind of national protection available): AE, AG, AL, AM, AO, AT, AU, AZ, BA, BB, BG, BH, BN, BR, BW, BY, BZ, CA, CH, CL, CN, CO, CR, CU, CZ, DE, DJ, DK, DM, DO, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT, HN, HR, HU, ID, IL, IN, IR, IS, JO, JP, KE, KG, KH, KN, KP, KR, KW, KZ, LA, LC, LK, LR, LS, LU, LY, MA, MD, ME, MG, MK, MN, MW, MX, MY, MZ, NA, NG, NI, NO, NZ, OM, PA, PE, PG, PH, PL, PT, QA, RO, RS, RU, RW, SA, SC, SD, SE, SG, SK, SL, SM, ST, SV, SY, TH, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, ZA, ZM, ZW.

(84) Designated States (unless otherwise indicated, for every kind of regional protection available): ARIPO (BW, GH, GM, KE, LR, LS, MW, MZ, NA, RW, SD, SL, ST, SZ, TZ, UG, ZM, ZW), Eurasian (AM, AZ, BY, KG, KZ, RU, TJ, TM), European (AL, AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HR, HU, IE, IS, IT, LT, LU, LV, MC, MK, MT, NL, NO, PL, PT, RO, RS, SE, SI, SK, SM,

(54) Title: SYSTEM AND METHOD FOR DETECTING PERSON ACTIVITY IN VIDEO

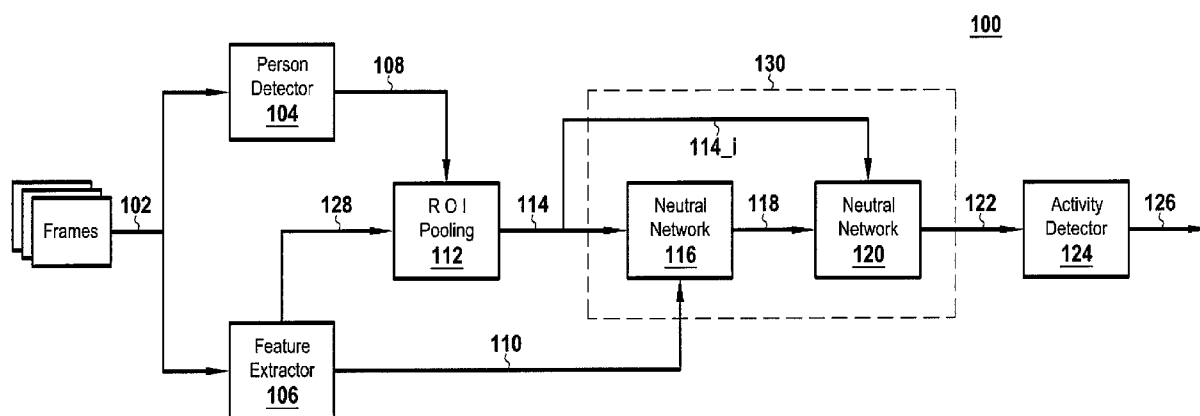


FIG.1

(57) Abstract: A system and a method for detecting person activity in a video are disclosed. The method comprises detecting one or more persons in a frame of the video; generating a first set of feature vectors based respectively on a set of frames of the video; generating a second set of person-specific feature vectors based on the set of frames; determining a temporal scene context vector for the frame, based on the first set of feature vectors and the second set of person-specific feature vectors; and for a detected person of the one or more detected persons, determining a hidden state vector for the frame, based on the temporal scene context vector and a subset of the second set of person-specific feature vectors corresponding to the detected person; and detecting one or more activities performed by the detected person in the frame based on the determined hidden state vector.



TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW,
KM, ML, MR, NE, SN, TD, TG).

Published:

— *with international search report (Art. 21(3))*

SYSTEM AND METHOD FOR DETECTING PERSON ACTIVITY IN VIDEO

Field of the disclosure

The present disclosure relates to the field of person activity
5 detection in a video.

Description of Related Art

The task of person activity detection in a video concerns
recognizing the activities of persons appearing in frames of the video
10 where each person can perform one or more activities in a frame.

A common conventional approach includes detecting bounding
boxes in each frame using object detectors and linking the detected
bounding boxes to obtain action tubes, which are then classified. Works
describing aspects of this conventional approach include "Gkiozari *et al.*,
15 "Finding action tubes," in *CVPR*, 2015, pp. 759-768," "Hou *et al.*, "Tube
Convolutional Neural Network (T-CNN) for Action Detection in Videos," in
ICCV, 2017, pp. 5822-5831," "Kalogeiton *et al.*, "Action Tubelet Detector
for Spatio-Temporal Action Localization, " in *ICCV*, 2017, pp. 4415-4423,"
"Singh *et al.*, "Online Real-Time Multiple Spatiotemporal Action
20 Localisation and Prediction," in *ICCV*, 2017, pp. 3657-3666," "Ross
Girshick, "Fast R-CNN," in *ICCV*, 2015, pp. 1440-1448," and "Chao *et al.*,
"Rethinking the Faster R-CNN Architecture for Temporal Action
Localization," in *CVPR*, 2018, pp. 1130-1139."

A deficiency of this conventional approach is that in order for it to
25 learn an underlying model (using full supervision) a fully-annotated
training dataset is required. This means that for every frame of the
training dataset all actions (activities) must be labeled and all persons
must be identified with bounding boxes. This is very expensive to
implement in practice. For example, the large-scale AVA 2.1 dataset

provides only sparse annotations at the rate of one frame per second (i.e., every second, only one frame of the video is annotated).

Other conventional approaches may be trained with sparsely annotated datasets (e.g., one annotated frame per second). These approaches are described in "Weinzaepfel *et al.*, "Towards weakly-supervised action localization," *arXiv: 1605.05197*, 2016," "Li *et al.*, "Recurrent tubelet proposal and recognition networks for action detection," in *ECCV*, 2018, pp. 303-318," and "Girdhar *et al.*, "A better baseline for AVA," *arXiv:1807.10066*, 2018." However, these methods treat each person in the frame independently, i.e., a person's activity is inferred without regard to potential interactions between persons in the frame which may or may not impact the person's activity.

In the context of group activity analysis, the relations between various individual persons are used to infer an action label for the group as well as for the individuals. Typically, these methods rely on a hierarchical model in which the individuals' actions are modeled in a lower level model and the group activity is modeled in an upper level model. Works describing these methods include in "Ibrahim *et al.*, "A Hierarchical Deep Temporal Model for Group Activity Recognition," in *CVPR*, 2016, pp. 1971-1980," "Biswas *et al.*, "Structural Recurrent Neural Network (SRNN) for Group Activity Analysis," in *WACV*, 2018, pp. 1625-1632," and "Tang *et al.*, "Mining Semantics-Preserving Attention for Group Activity Recognition," in *ACM-MM*, 2018, pp. 1283-1291."

However, these methods assume that each person is a part of a group (e.g., a sports team) and that each person performs a single action as part of the group, i.e., that the person's activity necessarily depends on being part of the group. As such, these methods are not suitable for videos in which a person may perform multiple activities in a frame and in

which the person's activities may or may not be based on interactions with other persons.

Summary

5 The present disclosure overcomes one or more deficiencies of the prior art by proposing a method for detecting person activity in a video, comprising:

 detecting one or more persons in a frame of the video;

 generating a first set of feature vectors based respectively on a set
10 of frames of the video, the set of frames including the frame and contained in a frame window centered at the frame;

 generating a second set of person-specific feature vectors based on the set of frames;

 determining, using a first trained neural network, a temporal scene
15 context vector for the frame, based on the first set of feature vectors and the second set of person-specific feature vectors; and

 for a detected person of the one or more detected persons,

 determining, using a second trained neural network, a
 hidden state vector for the frame, based on the temporal scene context
20 vector and a subset of the second set of person-specific feature vectors corresponding to the detected person; and

 detecting one or more activities performed by the detected person in the frame based on determined the hidden state vector.

 Accordingly, the one or more activities detected for a person are
25 identified as a function of a temporal scene context vector, which describes the temporal scene taking place around the frame being processed. The temporal scene context is defined by a frame window which size may be configured as desired. For example, the size of the frame window may be set depending on the activities that are most

sought to be detected in the video (certain actions can be better recognized with a larger temporal context than other actions).

The one or more activities detected for the person are also identified as a function of a hidden state vector, which contains implicit
5 information describing what is happening in the scene from the perspective of the detected person.

In an embodiment, detecting the one or more persons in the frame comprises determining one or more bounding boxes corresponding to the locations of one or more persons in the frame. In an embodiment, the one
10 or more bounding boxes are determined using a Faster Region-Based Convolutional Neural Network (FRCNN). In an embodiment, the FRCNN has a Residual Network (ResNet) architecture.

In an embodiment, the frame window comprises $L = 2l + 1$ frames, where $l > 0$.

15 In an embodiment, generating the first set of feature vectors comprises generating a first set of feature maps based on the L frames contained in the frame window; and generating the first set of feature vectors from the first set of feature maps.

In an embodiment, generating the first set of feature maps
20 comprises extracting Inflated 3D (I3D) feature maps from the L frames contained in the frame window.

Each feature vector of the second set of person-specific feature vectors includes person-specific feature information corresponding to at least one person detected in the respective frame corresponding to the
25 feature vector.

In an embodiment, generating the second set of person-specific feature vectors comprises extracting a second set of feature maps from the first set of feature maps; and generating the second set of person-specific feature vectors from the second set of feature maps.

In an embodiment, extracting the second set of feature maps from the first set of feature maps comprises, for each feature map of the first set of feature maps (which corresponds to a respective frame in the frame window), applying Region of Interest (RoI) pooling to the feature map
5 based on one or more bounding boxes detected in the respective frame.

In an embodiment, the first trained neural network is a recurrent neural network (RNN).

In an embodiment, determining the temporal scene context vector for the frame comprises:

- 10 performing a maxpool operation based on the second set of person-specific feature vectors to generate a maxpooled person feature vector;
concatenating the maxpooled person feature vector with the first set of feature vectors to generate a concatenated feature vector; and
applying the concatenated feature vector together with temporal
15 context vectors of a preceding frame and a consecutive frame to a bidirectional Gated Recurrent Unit (BiGRU) to generate the temporal scene context vector for the frame.

The maxpool operation based on the second set of person-specific feature vectors allows for the different person-specific feature vectors to
20 be combined in a compact manner, which reduces redundant information. The resulting maxpooled feature vector is relatively small in size, which increases the speed of determining the temporal scene context vector.

In an embodiment, determining the hidden state vector for the frame comprises determining the hidden state vector based on the
25 temporal scene context vector, the subset of the second set of person-specific feature vectors corresponding to the detected person, and a combined hidden state vector corresponding to the one or more detected persons in the frame.

The combined hidden state vector contains information representative of relations between potential activities performed by the one or more detected persons in the frame. As such, the one or more activities detected for the person are identified additionally based on
5 interactions taking place between persons in the frame. In other words, the detection also takes into account whether a particular interaction impacts or not the person's activity and determines the person's activity accordingly.

In an embodiment, the second trained neural network is a
10 recurrent neural network (RNN).

In an embodiment, determining the hidden state vector for the frame comprises:

performing a maxpool operation on previous iteration hidden state vectors corresponding respectively to the one or more detected persons in
15 the frame to generate the combined hidden state vector;

concatenating the combined hidden state vector with the temporal scene context vector and the subset of the second set of person-specific feature vectors corresponding to the detected person to generate a concatenated vector; and

20 applying the concatenated vector to a Gated Recurrent Unit (GRU) to generate the hidden state vector corresponding to the frame.

In an embodiment, detecting the one or more activities performed by the detected person in the frame comprises:

applying the determined hidden state vector to a fully-connected
25 layer to generate a set of probabilities, each probability corresponding to a respective activity of a set of activities and indicating the probability that the detected person is performing the respective activity in the frame; and
detecting the one or more activities performed by the detected person in the frame based on the generated set of probabilities.

In an embodiment, applying the determined hidden state vector to the fully-connected layer comprises applying the hidden state vector to the fully-connected layer together with equivalent hidden state vectors corresponding to other detected persons in the frame. As such, the
5 detection of the one or more activities performed by the detected person benefit from information about the interactions taking place between the different actors in the video. As certain activities involve interaction between people (e.g., looking at someone, talking to someone, etc.) and other activities don't (e.g., sitting, standing, etc.), the interaction
10 information helps distinguish at least between these two categories of activities.

In an embodiment, detecting the one or more activities for the detected person in the frame comprises selecting one or more activities from the set of activities having respective probabilities above a threshold.
15 The threshold may be configurable which allows for increasing or decreasing the confidence level with which activities are selected.

The present disclosure also proposes a system for detecting person activity in a video, comprising:

a person detector configured to detect one or more persons in a
20 frame of the video;

a feature extractor configured to generate a first set of feature vectors based respectively on a set of frames of the video, the set of frames including the frame and contained in a frame window centered at the frame;

25 a Region of Interest (RoI) pooling module configured to generate a second set of person-specific feature vectors based on the set of frames;

a first trained neural network configured to determine a temporal scene context vector for the frame, based on the first set of feature vectors and the second set of person-specific feature vectors;

a second trained neural network configured to determine a hidden state vector for the frame for a detected person of the one or more detected persons, based on the temporal scene context vector and a subset of the second set of person-specific feature vectors corresponding to the detected person; and

an activity detector configured to detect one or more activities performed by the detected person in the frame based on the determined hidden state vector.

This system may be configured to perform any of the above-described method embodiments of to the present disclosure.

In an embodiment, any of the above-described method embodiments may be implemented as instructions of a computer program. As such, the present disclosure provides a computer program including instructions that when executed by a processor cause the processor to execute a method according to any of the above-described method embodiments.

The computer program can use any programming language and may take the form of a source code, an object code, or a code intermediate between a source code and an object code, such as a partially compiled code, or any other desirable form.

The computer program may be recorded on a computer-readable medium. As such, the present disclosure is also directed to a computer-readable medium having recorded thereon a computer program as described above. The computer-readable medium can be any entity or device capable of storing the computer program.

Brief description of the drawings

Further features and advantages of the present disclosure will become apparent from the following description of certain embodiments

thereof, given by way of illustration only, not limitation, with reference to the accompanying drawings in which:

FIG. 1 illustrates a system for detecting person activity in a video according to an embodiment;

5 FIG. 2 illustrates an example computer implementation of the system of FIG. 1;

FIG. 3 illustrates an example implementation of a first neural network according to an embodiment;

10 FIG. 4 illustrates an example implementation of a second neural network according to an embodiment; and

FIG. 5 illustrates an example implementation of an activity detector according to an embodiment.

Description of embodiments

15 FIG. 1 illustrates a system 100 for detecting person activity in a video according to an embodiment. System 100 is provided for the purpose of illustration only and is not limiting of embodiments of the present disclosure.

20 As shown in FIG. 1, system 100 includes a person detector 104, a feature extractor 106, a Region of Interest (ROI) pooling module 112, a hierarchical neural network 130, and an activity detector 124.

In an embodiment, the hierarchical neural network 130 includes a first trained neural network 116 and a second trained neural network 120. The first trained neural network 116 and the second trained neural
25 network 120 may be trained together or separately during a training phase. Only a sparsely-annotated training dataset (e.g., one annotated frame or less per second) is required to train the first neural network 116 and the second neural network 120. This is due to the fact that neural network 130 is configured to operate on the basis of a frame window of

several video frames as further described below. In an embodiment, the AVA 2.1 dataset is used to train the first neural network 116 and the second neural network 120. The AVA 2.1 dataset is annotated for 60 action classes (i.e., 60 person activities) and includes 235 videos of 15 minutes each, annotated at the rate of one frame per second.

In an embodiment, system 100 may be implemented on a computer system such as computer system 200 shown in FIG. 2. Specifically, system 100 may be implemented as a computer program including instructions that, when executed by a processor 202 of computer system 200, cause the processor 202 to execute methods or functions of system 100 as further described herein. In an embodiment, the computer program may be recorded on a computer-readable medium 204 of computer system 200.

Returning to FIG. 1, system 100 is configured to receive a video including a series of frames 102. It is noted that a frame corresponds to a single image of the video. As further described below, for a given frame (hereinafter referred to as "the frame" or the "frame t") of the video, system 100 detects one or more activities for at least one detected person in the frame (assuming that a person is detected in the frame). The processing as further described herein is repeated per frame.

As shown in FIG. 1, the frames 102 are provided to person detector 104 and feature extractor 106.

Person detector 104 acts on the frames 102 on a frame-by-frame basis. Specifically, for a frame t of the video, person detector 104 is configured to detect one or more persons in the frame.

In an embodiment, detecting the one or more persons in the frame comprises determining one or more bounding boxes corresponding to the locations of one or more persons in the frame. In an embodiment, the one or more bounding boxes are determined using a Faster Region-Based

Convolutional Neural Network (FRCNN) as described in "Ren *et al.*, "Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks," in *NIPS*, 2015, pp. 991-99). The FRCNN may have a Residual Network (ResNet) architecture as described in "He *et al.*, "Deep Residual Learning for Image Recognition," in *CVPR*, 2016, pp. 770-778).

Before being used as part of system 100, person detector 104 is trained to detect persons in frames. In an embodiment, person detector 104 is trained on a generic object dataset (e.g., PASCAL Visual Object Classes (VOC)) and then fine-tuned for improved person detection.

Assuming that one or more persons are detected in the frame t by person detector 104, the processing of the frame t by system 100 continues. Otherwise, a next frame is processed. Hereinafter, it is assumed that one or more persons are detected in the frame t .

Feature extractor 106 is configured to extract features that can be used to describe the temporal scene context of the frame t . The temporal scene context of the frame t describes the scene that is taking place around the frame t . For example, a series of frames may show a person opening the door of a car, entering the car, and sitting behind the wheel. The temporal scene context for a frame in this series of frames is then a person entering into a car.

In an embodiment, feature extractor 106 is pre-configured to define a frame window centered at the frame t based on which the temporal scene context of the frame t is to be determined. In an embodiment, the frame window comprises $L = 2l + 1$ frames, where $l > 0$. For example, the frame window may include 3 frames, i.e., the frame t , a frame $t-1$, and a frame $t+1$. The frames $t-1$, t , and $t+1$ may or may not be immediately consecutive within the series of frames, e.g., the frame $t-1$ may precede the frame t by one second, and the frame $t+1$ may follow

the frame t by one second, but intervening frames may be present between the frames t and $t-1$ and the frames t and $t+1$.

In an embodiment, feature extractor 106 is configured to generate a first set of feature vectors 110 based respectively on a set of frames of the video. The set of frames include the frame t and are contained in a frame window centered at the frame t . Assuming a frame window of size $L=2l+1$, i.e., l frames before and l frames after the frame t , the first set of features vectors 110 includes L feature vectors. Hereinafter, the L feature vectors are denoted mathematically by the vector sequence $\langle x_c^t \rangle = x_c^{t-l}, \dots, x_c^{t-1}, x_c^t, x_c^{t+1}, \dots, x_c^{t+l}$. Assuming that the L frames are separated by one second, the L features vectors correspond to a temporal scene context of L seconds.

In an embodiment, generating the first set of feature vectors 110 comprises generating a first set of feature maps based on the L frames contained in the frame window; and generating the first set of feature vectors 110 from the first set of feature maps. In an embodiment, the first set of feature maps are flattened to a single dimension to obtain the corresponding first set of feature vectors 110.

In an embodiment, a single feature map, and feature vector, is generated per frame of the frame window.

A feature map contains spatio-temporal information that captures what can be viewed in the frame (e.g., objects and activities appearing in the frame). Typically, a feature map may include one or more features. Each feature may be computed based on a temporal context of several frames that surround the frame for which the feature map is computed.

In an embodiment, generating the first set of feature maps comprises extracting Inflated 3D (I3D) feature maps from the L frames contained in the frame window. This may be done as described in

"Carreira *et al.*, "Quo Vasis, Action recognition? A new model and the kinetics dataset," in *CVPR*, 2017, pp. 4724-4733".

Returning to FIG. 1, ROI pooling module 112 is configured to generate a second set of person-specific feature vectors 114 based on the set of frames contained in the frame window. Each feature vector of the second set of person-specific feature vectors 114 includes person-specific feature information corresponding to a respective person detected in the respective frame corresponding to the feature vector.

In an embodiment, ROI pooling module 112 receives from person detector 104 a dataset 108 including, for each frame of the L frames contained in the frame window the one or more bounding boxes determined for the frame (corresponding to the one or more persons detected in the frame). In addition, ROI pooling module 112 receives from feature extractor 106 a dataset 128 containing the first set of feature maps generated by feature extractor 106 based on the L frames contained in the frame window.

In an embodiment, generating the second set of person-specific feature vectors 114 comprises extracting a second set of feature maps from the first set of feature maps; and generating the second set of person-specific feature vectors 114 from the second set of feature maps.

In an embodiment, extracting the second set of feature maps from the first set of feature maps comprises, for each feature map of the first set of feature maps (which corresponds to a respective frame in the frame window), applying Region of Interest (RoI) pooling to the feature map based on the one or more bounding boxes detected in the respective frame (obtained from dataset 108). Specifically, the one or more bounding boxes detected in the respective frame are used to pull specific feature map information from the feature map (of the first set) corresponding to the frame, the specific feature map information being

the spatio-temporal information specific to the person(s) detected in the frame.

In an embodiment, the second set of feature maps are extracted from the first set of feature maps 128 as described in "Gu *et al.*, "AVA: A
5 Video dataset of Spatio-Temporally Localized Atomic Visual Actions," in *CVPR*, 2018, pp. 6047-6056".

In an embodiment, for each frame contained in the frame window, one or more person-specific feature vectors corresponding respectively to one or more persons detected in the frame are generated. Generally, the
10 persons detected in the set of L frames corresponding to the frame window are substantially the same over the set of frames, such that a detected person p_i is associated with L person-specific feature vectors each corresponding to a respective frame of the L frames.

Hereinafter, a person-specific feature vector corresponding to a
15 detected person p_i in frame t is denoted by $x_{p_i}^t$.

Returning to FIG. 1, the first neural network 116 is configured to receive the first set of L feature vectors 110 from feature extractor 106 and the second set of person-specific feature vectors 114 from ROI pooling module 112. The first neural network 116 is configured to
20 determine a temporal scene context vector 118 for the frame t , based on the first set of feature vectors 110 and the second set of person-specific feature vectors 114. In an embodiment, the first neural network is a recurrent neural network (RNN).

FIG. 3 illustrates an example implementation 300 of the first neural
25 network 116 according to an embodiment. As shown in FIG. 3, example first neural network 300 includes a maxpool module 302, a concatenation module 306, and a bidirectional Gated Recurrent Unit (BiGRU) 310.

The maxpool module 302 is configured to perform a maxpool operation based on the second set of person-specific feature vectors 114 to generate a maxpooled person feature vector 304.

In an embodiment, the maxpooled person feature vector 304 is
 5 computed as $x_p^t = \maxpool_i(x_{p_i}^t)$, $i = 1, \dots, V$ ($V > 0$). In other words, the maxpooled person feature vector 304 is obtained by maxpooling the person-specific feature vectors $x_{p_i}^t$ corresponding to all detected persons (V) in the frame t.

The concatenation module 306 is configured to concatenate the
 10 maxpooled person feature vector 304 with the first set of L feature vectors to generate a concatenated feature vector 308.

In an embodiment, the concatenated feature vector 308, denoted is computed as $x^t = \langle x_c^t \rangle \oplus x_p^t$, where $\oplus$ denotes vector concatenation.

The concatenated feature vector 308 is applied together with
 15 temporal context vectors 312 of a preceding frame (t-1) and a consecutive frame (t+1) to BiGRU 310 to generate the temporal scene context vector 118 for the frame t. A BiGRU is a recurrent network in which temporal nodes are connected bidirectionally such that information from one node in the temporal sequence affects the adjacent nodes (t-1, t+1). A
 20 description of a BiGRU can be found, for example, in "Schuster, Mike and Kuldeep K. Paliwal, "Bidirectional recurrent neural network," *IEEE Transactions on Signal Processing*, Vol. 45, No. 11, (1997): 2673-2681".

In an embodiment, the temporal scene context vector 118 is computed as $h_c^t = biGRU(x^t; h_c^{t-1}; h_c^{t+1})$, where h_c^{t-1} and h_c^{t+1} denote the
 25 temporal scene context vectors 312 of the preceding frame (t-1) and the consecutive frame (t+1).

Returning to FIG. 1, the second neural network 120 is configured to receive the temporal scene context vector 118 of the frame t and the second set of person-specific feature vectors 114. For each detected

person i in the frame t , the second neural network 120 is configured to determine a hidden state vector 122 for the frame t , based on the temporal scene context vector 118 and a subset 114 _{i} of the second set of person-specific feature vectors 114 corresponding to the detected
5 person i . The hidden state vector 122 for a detected person contains implicit information describing what is happening in the scene from the perspective of the detected person.

In an embodiment, the second neural network 120 is a recurrent neural network (RNN).

10 In an embodiment, given V detected person in the frame t , the second neural network 120 models the relations of the detected persons and uses this modeling to infer the activity(ies) of each of them. With each detected person represented as a node $v_i \in V$ of a fully-connected graph, the second neural network 120 iteratively updates a hidden state
15 vector for each node v_i based on the intermediate hidden state vectors of the other nodes.

In an embodiment, the hidden state vector 122 for a detected person i is determined based on the temporal scene context vector 118, the subset 114 _{j} of the second set of person-specific feature vectors
20 corresponding to the detected person, and a combined hidden state vector corresponding to the one or more detected persons in the frame.

In an embodiment, the combined hidden state vector contains information representative of relations between potential activities performed by the one or more detected persons in the frame. As such, the
25 one or more activities detected for a detected person (which as described below are determined based on the hidden state vector 122) are identified additionally based on interactions taking place between persons in the frame. In other words, the activity detection also takes into account whether a particular interaction impacts or not the person's activity and

determines the person's activity accordingly. This is based on the insight that actions of different persons may be correlated (e.g., "talk to" and "listen to", actions often performed by a group together, e.g., "play instrument"), uncorrelated, or even exclusive of each other (e.g., if a person "drives" a car, it is unlikely that other persons in the car "stand" or "play an instrument").

FIG. 4 illustrates an example implementation 400 of the second neural network 120 according to an embodiment. As shown in FIG. 4, example second neural network 400 includes a maxpool module 402, a concatenation module 408, and a Gated Recurrent Unit (GRU) 412.

The maxpool module 402 is configured to perform a maxpool operation on previous iteration hidden state vectors 402, which correspond respectively to the one or more detected persons in the frame t , to generate a combined hidden state vector 406. In an embodiment, the combined hidden state vector 406 for iteration j is computed as $x_{v_i}^{(j)} = \text{maxpool}_{v_i \in V} (h_{v_i}^{(j-1)})$, where $h_{v_i}^{(j-1)}$ denotes the hidden state vector of node v_i for iteration $(j-1)$ (for frame t).

The combined hidden state vector 406 is then concatenated, using concatenation module 408, with the temporal scene context vector 118, the subset 114_i of the second set of person-specific feature vectors corresponding to the detected person i to generate a concatenated vector 410. In an embodiment, the concatenated vector 410 for detected person i (for frame t) for iteration j is computed as $a_{v_i}^{(j)} = x_{p_i}^t \oplus h_c^t \oplus x_{v_i}^{(j)}$.

The concatenated vector 410 is then applied to GRU 412 to generate the hidden state vector 122 for the detected person i for the frame t .

In an embodiment, the hidden state vector 122 for the detected person i for iteration j (for the frame t), denoted as $h_{v_i}^{(j)}$, is computed using the following operations (which are computed inside GRU 412):

$$\begin{aligned}
z_{v_i}^{(j)} &= \sigma(U^z a_{v_i}^{(j)} + W^z h_{v_i}^{(j-1)}) \\
r_{v_i}^{(j)} &= \sigma(U^r a_{v_i}^{(j)} + W^r h_{v_i}^{(j-1)}) \\
s &= \tanh(U^s a_{v_i}^{(j)} + W^s (h_{v_i}^{(j-1)} \circ r_{v_i}^{(j-1)})) \\
h_{v_i}^{(j)} &= (1 - z_{v_i}^{(j)}) \circ h_{v_i}^{(j-1)} + z_{v_i}^{(j)} \circ s
\end{aligned}$$

where σ represents the sigmoid function, U^z , W^z , U^r , W^r , U^s , and W^s denote weights of GRU 412, and $\circ$ represents the Hadamard product.

Typically, after a fixed number of iterations, the hidden state vector 122 for the detected person i is output by GRU 412.

5 Returning to FIG. 1, activity detector 124 is configured to receive the hidden state vector 122 for detected person i (for frame t) from the second neural network 120. Activity detector 124 is configured to detect one or more activities 126 performed by the detected person i based on the hidden state vector 122.

10 FIG. 5 illustrates an example implementation 500 of the activity detector 124 according to an embodiment. As shown in FIG. 5, example activity detector 500 includes a fully-connected layer 502 and an activity inference module 506.

15 Fully-connected layer 502 is configured to receive the hidden state vector 122 for a detected person i and to generate a set of probabilities 504_1, ..., 504_M each corresponding to a respective activity of a predefined set of activities (e.g., 60 possible activities) and indicating the probability that the detected person is performing the respective activity in the frame t .

20 In an embodiment, the fully-connected layer 502 additionally receives the equivalent hidden state vectors for frame t of some or all of the other detected persons in the frame t , and calculates the set of probabilities 504_1, ..., 504_M with account to all received hidden state vectors. As such, the set of probabilities 504_1, ..., 504_M reflect that the

action(s) performed by the detected person *i* in frame *t* depend, at least in part, on the actions of the other persons detected in frame *t*.

A fully-connected layer is a known neural network component. A description of fully-connected layers can be found for example in "LeCun, Yann, *et al.*, "Backpropagation applied to handwritten zip code recognition," *Neural computation* 1.4 (1989): 541-551".

In an embodiment, the fully-connected layer 502 has a sigmoid as activation function.

Based on the generated set of probabilities (504_1, ..., 504_M), activity inference module 506 is configured to detect the one more activities 126 performed by the detected person *i* in the frame *t*.

In an embodiment, the activities having a respective probability above a predefined threshold are taken as the one or more detected activities 126 of the detected person *i*. The threshold may be set according to a desired confidence level in the activities detected.

In another embodiment, the thresholding may be further combined with inferences based on what other detected persons are doing. For example, if the probability that detected person *i* is speaking is 0.5 (i.e., 50%) and the probability that detected person *j* is speaking is 0.3, then it is likely that the person *i* is speaking and person *j* is listening in the frame.

Although the present invention has been described above with reference to certain specific embodiments, it will be understood that the invention is not limited by the particularities of the specific embodiments. Numerous variations, modifications and developments may be made in the above-described embodiments within the scope of the appended claims.

CLAIMS

1. A method for detecting person activity in a video, comprising:
 - 5 detecting (104) one or more persons in a frame of the video;
generating (106) a first set of feature vectors (110) based respectively on a set of frames of the video, the set of frames including the frame and contained in a frame window centered at the frame;
generating (112) a second set of person-specific feature vectors
10 (114) based on the set of frames;
determining, using a first trained neural network (116), a temporal scene context vector (118) for the frame, based on the first set of feature vectors (110) and the second set of person-specific feature vectors (114);
and
15 for a detected person of the one or more detected persons,
determining, using a second trained neural network (120), a hidden state vector (122) for the frame, based on the temporal scene context vector (118) and a subset (114_i) of the second set of person-specific feature vectors corresponding to the detected person; and
20 detecting (124) one or more activities (126) performed by the detected person in the frame based on the determined hidden state vector (122).
2. The method of claim 1, wherein determining the temporal
25 scene context vector (118) for the frame comprises:
performing a maxpool operation (302) based on the second set of person-specific feature vectors (114) to generate a maxpooled person feature vector (304);

concatenating (306) the maxpooled person feature vector (304) with the first set of feature vectors (110) to generate a concatenated feature vector (308); and

5 applying the concatenated feature vector (308) together with temporal context vectors (312) of a preceding frame and a consecutive frame to a bidirectional Gated Recurrent Unit, BiGRU (310) to generate the temporal scene context vector (118) for the frame.

3. The method of any of claims 1 and 2, wherein determining
10 the hidden state vector (122) for the frame comprises determining the hidden state vector (122) based on the temporal scene context vector (118), the subset (114_i) of the second set of person-specific feature vectors corresponding to the detected person, and a combined hidden state vector (406) corresponding to the one or more detected persons in
15 the frame.

4. The method of claim 3, wherein the combined hidden state vector (406) contains information representative of relations between potential activities performed by the one or more detected persons in the
20 frame.

5. The method of any of claims 3 and 4, wherein determining the hidden state vector (122) for the frame comprises:

25 performing a maxpool operation (404) on previous iteration hidden state vectors (402) corresponding respectively to the one or more detected persons in the frame to generate the combined hidden state vector (406);

concatenating (408) the combined hidden state vector (406) with the temporal scene context vector (118) and the subset (114_i) of the

second set of person-specific feature vectors corresponding to the detected person to generate a concatenated vector (410); and

applying the concatenated vector (410) to a Gated Recurrent Unit, GRU (412) to generate the hidden state vector (122) corresponding to the
5 frame.

6. The method of claim 1, wherein detecting (124) the one or more activities (126) performed by the detected person in the frame comprises:

10 applying the determined hidden state vector (122) to a fully-connected layer (502) to generate a set of probabilities (504_1, ..., 504_M), each probability corresponding to a respective activity of a set of activities and indicating the probability that the detected person is performing the respective activity in the frame; and

15 detecting (506) the one or more activities (126) performed by the detected person in the frame based on the generated set of probabilities (504_1, ..., 504_M).

7. The method of claim 6, wherein applying the determined
20 hidden state vector (122) to the fully-connected layer (502) comprises applying the hidden state vector (122) to the fully-connected layer together with equivalent hidden state vectors corresponding to other detected persons in the frame.

25 8. A system (100) for detecting person activity in a video, comprising:

a person detector (104) configured to detect one or more persons in a frame of the video;

a feature extractor (106) configured to generate a first set of feature vectors (11) based respectively on a set of frames of the video, the set of frames including the frame and contained in a frame window centered at the frame;

- 5 a Region of Interest, RoI pooling module (112) configured to generate a second set of person-specific feature vectors (114) based on the set of frames;

 a first trained neural network (116) configured to determine a temporal scene context vector (118) for the frame, based on the first set
10 of feature vectors (110) and the second set of person-specific feature vectors (114);

 a second trained neural network (120) configured to determine a hidden state vector (122) for the frame for a detected person of the one or more detected persons, based on the temporal scene context vector
15 (118) and a subset (114_i) of the second set of person-specific feature vectors corresponding to the detected person; and

 an activity detector (124) configured to detect one or more activities (126) performed by the detected person in the frame based on the determined hidden state vector (122).

20

9. A computer program (PROG) including instructions that when executed by a processor (202) cause the processor (202) to execute a method according to any one of claims 1 to 7.

25 10. A computer-readable medium (204) recorded thereon a computer program according to claim 9.

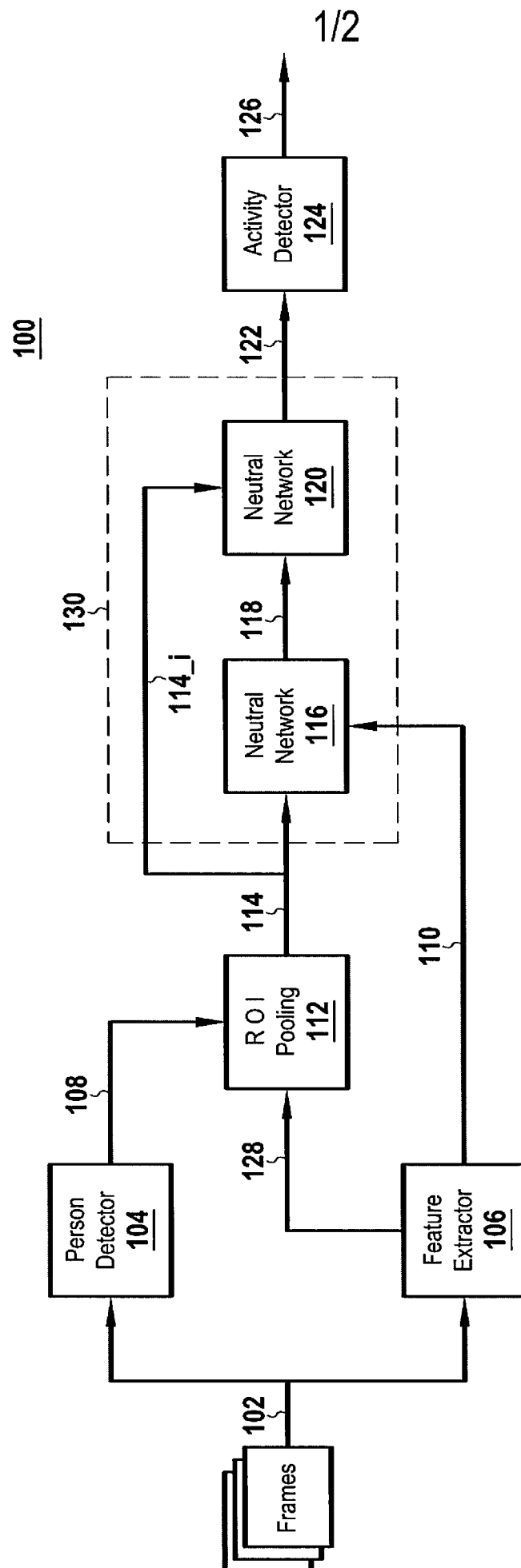


FIG.1

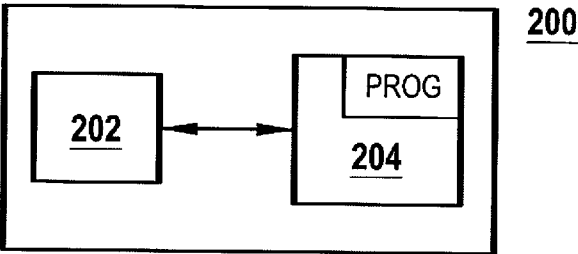


FIG.2

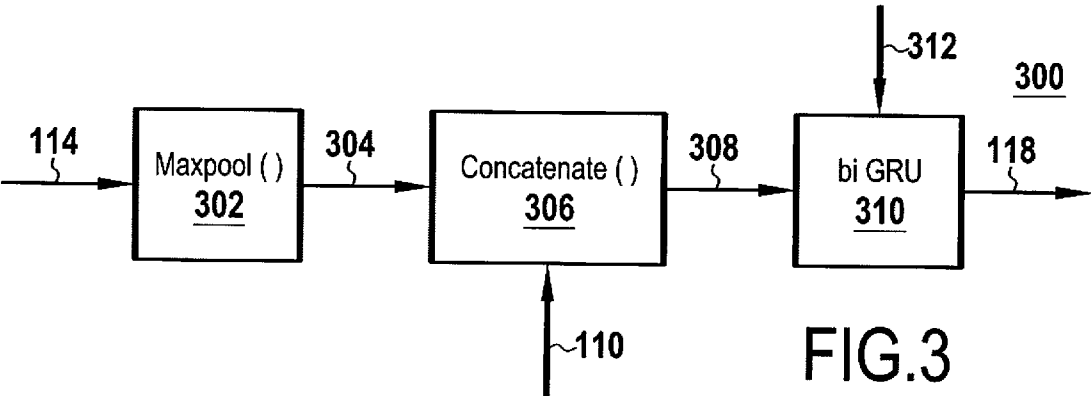


FIG.3

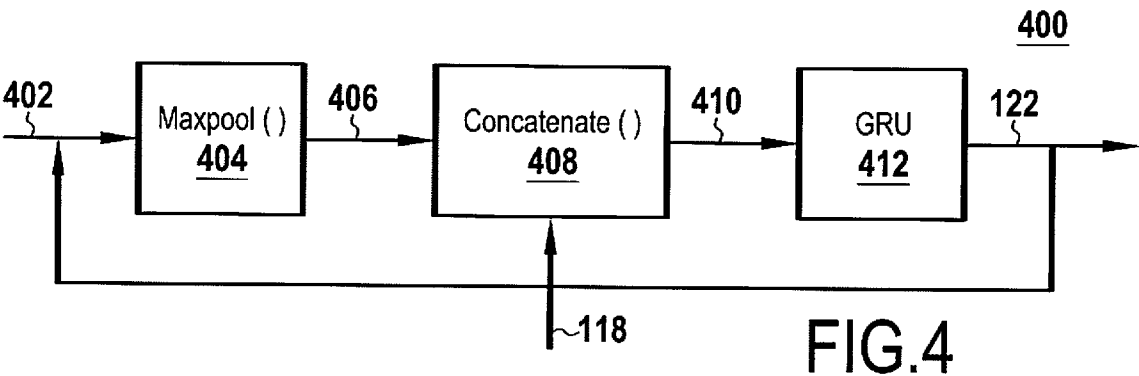


FIG.4

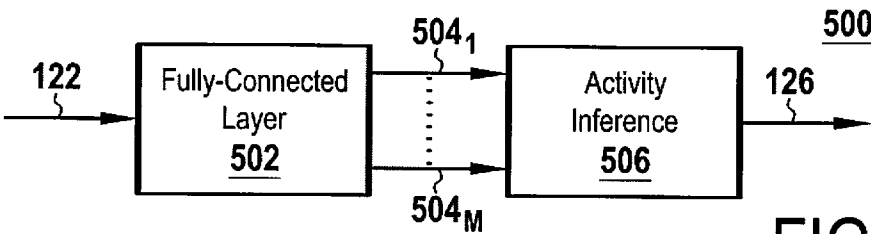


FIG.5

INTERNATIONAL SEARCH REPORT

International application No
PCT/EP2019/072348

A. CLASSIFICATION OF SUBJECT MATTER
INV. G06K9/00
ADD.

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)
G06K

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)

EPO-Internal, WPI Data

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	SINGH BHARAT ET AL: "A Multi-stream Bi-directional Recurrent Neural Network for Fine-Grained Action Detection", 2016 IEEE CONFERENCE ON COMPUTER VISION AND PATTERN RECOGNITION (CVPR), IEEE, 27 June 2016 (2016-06-27), pages 1961-1970, XP033021374, DOI: 10.1109/CVPR.2016.216 [retrieved on 2016-12-09] page 3, left column, lines 39-42 page 3, right column, lines 1-9 page 3, right column, lines 14-16 page 3, right column, lines 22-31; figure 2 page 4, left column, last paragraph; figure 3 page 4, right column, last paragraph page 5, right column, lines 14-21 page 5, right column, lines 22-27 -/--	1-10



Further documents are listed in the continuation of Box C.



See patent family annex.

* Special categories of cited documents :

"A" document defining the general state of the art which is not considered to be of particular relevance

"E" earlier application or patent but published on or after the international filing date

"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)

"O" document referring to an oral disclosure, use, exhibition or other means

"P" document published prior to the international filing date but later than the priority date claimed

"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention

"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone

"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art

"&" document member of the same patent family

Date of the actual completion of the international search

22 April 2020

Date of mailing of the international search report

04/05/2020

Name and mailing address of the ISA/

European Patent Office, P.B. 5818 Patentlaan 2
NL - 2280 HV Rijswijk
Tel. (+31-70) 340-2040,
Fax: (+31-70) 340-3016

Authorized officer

Natanni, Francesco

INTERNATIONAL SEARCH REPORT

International application No
PCT/EP2019/072348

C(Continuation). DOCUMENTS CONSIDERED TO BE RELEVANT		
Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
A	----- TKACHENKO DMYTRO: "Human Action Recognition Using Fusion of Modern Deep Convolutional and Recurrent Neural Networks", 2018 IEEE FIRST INTERNATIONAL CONFERENCE ON SYSTEM ANALYSIS & INTELLIGENT COMPUTING (SAIC), IEEE, 8 October 2018 (2018-10-08), pages 1-6, XP033433486, DOI: 10.1109/SAIC.2018.8516860 [retrieved on 2018-10-31] abstract page 3, left column, point "E. Recurrent neural network architecture" -----	2