Lec3-8-3-1-DeepLearningBehavior.ppt

Harry Li, Ph.D.

CTI One Corporation
3679 Enochs Street
Santa Clara, CA 95051

12-27-2018 3D CNN in Keras - Action Recognition

http://learnandshare645.blogspot.com/2016/06/3d-cnn-in-keras-action-recognition.html

#Reading boxing action class
#Reading hand clapping action class
#Reading hand waving action class
#Reading jogging action class
#Reading running action class
#Reading walking action class

https://www.youtube.com/watch?v=ecbeIRVqD7g&feature=youtu.be

Anuj shah
Published on Jun 19, 2016
This video explains the implementation of 3D CNN for action recognition. It explains little theory about 2D and 3D Convolution. The implementation of the 3D CNN in Keras continues in the next part. The codes are available at - http://learnandshare645.blogspot.in/2.... The links in the videos are given below : -

1. Install Keras from https://github.com/MinhazPalasara
   https://github.com/MinhazPalasara/ker...
2. Download KTH action dataset http://www.nada.kth.se/cvap/actions/
3. Configure opencv for video processing – install ffmeg and set up the path
   http://kronoskoders.logdown.com/

12-26-2018 Behavior Analysis

DEEP LEARNING BASED HUMAN BEHAVIOR RECOGNITION IN INDUSTRIAL WORKFLOWS

Konstantinos Makantasis 1
Anastasios Doulamis 2
Nikolaos Doulamis 2
Konstantinos Psychas 3

1, 2: Technical University of Crete, School of Production Engineering and Management, Chania, Greece; National Technical University of Athens, School of Electrical and Computer Engineering, Athens, Greece
3: Columbia University, Electrical Engineering Department, New York, USA

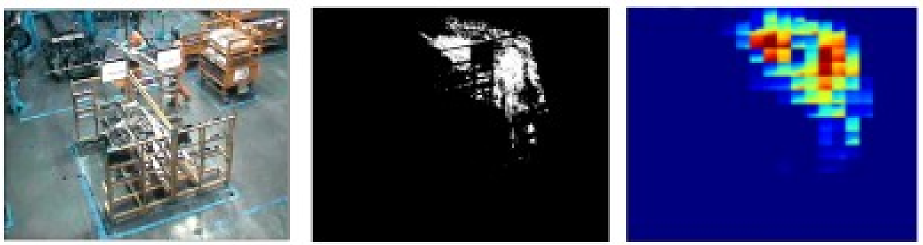
Rely in tutor to fusing spatio-temporal information of human activities into 2D maps before feding to a CNN, which, in turn, automatically constructs high-level features to capture the human activities.

Computational complexity is O(KLM log KLM) [15], with 2D convolution,O(KL log KL). Variables K and L stand for video's height and width, while M stands for the number of subsequent frames

The fully automated behavior understanding in industrial environments. we exploit a Convolutional Neural Network (CNN), which automate the process of feature construction. Although such models are limited to handle still 2D inputs, we transform video input to incorporate temporal information into each frame. Our model hierarchically constructs features from both spatial and temporal dimensions. Data taken from Nissan factory, and it achieves superior performance without relying on handcrafted features.

12-26-2018 DCT Feature Reduction

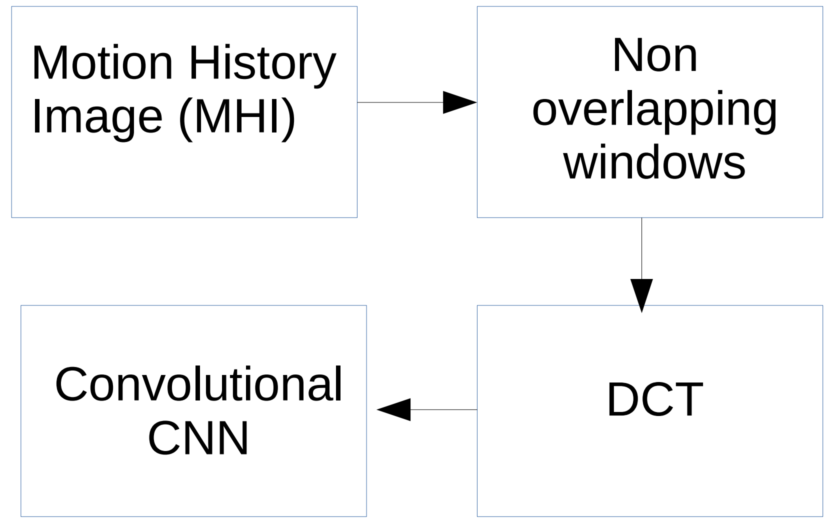
DEEP LEARNING BASED HUMAN BEHAVIOR RECOGNITION IN INDUSTRIAL WORKFLOWS



(a) (b) (c)

(a) Original frame,(b) MHI for the captured frame and (c) dimension reduction using DCT transform.

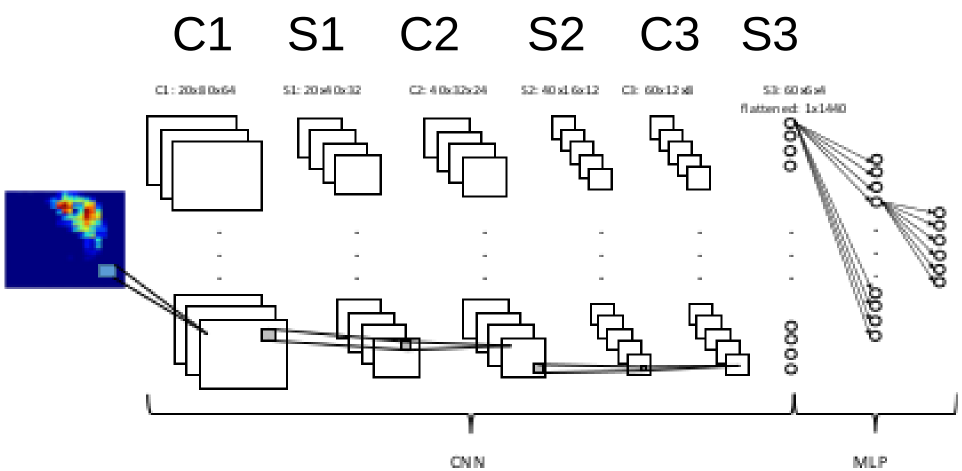
Incorporate temporal information into constructed features, we model the visual content of each frame using its Motion History Image (MHI) over a predefined number frames [17]. It has been shown in [16] that a small number of subsequent frames are enough to achieve an activity recognition performance.



To achieve a computational efficient, we split MHI input into non overlapping windows and keep the most dominant Discrete Cosine Transform (DCT) coefficients.

12-26-2018 CNN Architecture

DEEP LEARNING BASED HUMAN BEHAVIOR RECOGNITION IN INDUSTRIAL WORKFLOWS



12-26-2018 Experimental Setup

DEEP LEARNING BASED HUMAN BEHAVIOR RECOGNITION IN INDUSTRIAL WORKFLOWS

Deep learning tools

Theano [19, 20] library.

Data set library

The SCOVIS dataset consists of 20 scenarios.

Method	precision	recall
Echo State Networks	0.777	0.772
HMM-PF	0.797	0.788
MC-TDSVM (40% training set)	0.857	0.857
HMM-NN	0.875	0.863
MC-TDSVM (60% training set)	0.871	0.863
Our Approach (testing set)	**0.983**	**0.978**
Our Approach (unknown data)	**0.867**	**0.892**

Table 1. Other approaches

(i) Echo State Networks (ESN) [4], with handcrafted local motion grid features
(ii) Hidden Markov Model exploiting Particle Filters (HMM-PF) and Hidden Markov Model using a Neural Network rectification scheme (HMM-NN) [5], with handcrafted features using Zernike moments of pixel changing history [21]
(iii) MultiClass Tapped Delay Support Vector Machines (MC-TDSVM) [6] with handcrafted features [5] along with a user feedback strategy.

12-27-2018 SCOVIS

Self-Configurable Cognitive Video Supervision

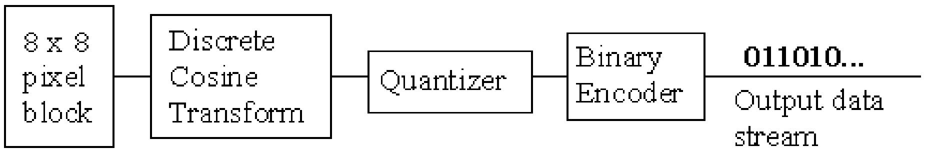
http://archive.li/Ec6qf

SCOVIS supports the automatic detection of
a) behaviours
b) workflow violation and
c) localization of salient moving or static objects in scenes, monitoring by multiple cameras (static or active).

Project Coordinator, Institute of Communication and Computer Systems/National Technical University of Athens (ICCS/NTUA)

12-27-2018 DCT Technique

http://www.johnloomis.org/ece563/notes/compression/jpeg/tutorial/jpegtut1.html



16	11	10	16	24	40	51	61
12	12	14	19	26	58	60	55
14	13	16	24	40	57	69	56
14	17	22	29	51	87	80	62
18	22	37	56	68	109	103	77
24	35	55	64	81	104	113	92
49	64	78	87	103	121	120	101
72	92	95	98	112	100	103	99

Quantization table for Luminance

17	18	24	47	99	99	99	99
18	21	26	66	99	99	99	99
24	26	56	99	99	99	99	99
47	66	99	99	99	99	99	99
99	99	99	99	99	99	99	99
99	99	99	99	99	99	99	99
99	99	99	99	99	99	99	99
99	99	99	99	99	99	99	99

Quantization table for chrominance

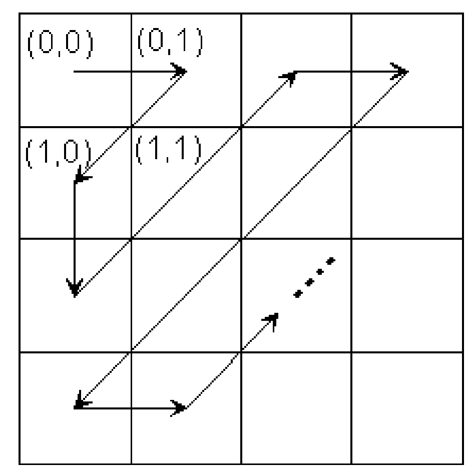
$$B(k_1,k_2) = \sum_{i=0}^{N_1-1} \sum_{j=0}^{N_2-1} 4\cdot A(i,j)\cdot\cos\left[\frac{\pi\cdot k_1}{2\cdot N_1}\cdot(2\cdot i+1)\right]\cdot\cos\left[\frac{\pi\cdot k_2}{2\cdot N_2}\cdot(2\cdot j+1)\right]$$

Quantization Matrix

The quantization matrix is the 8 by 8 matrix of step sizes (sometimes called quantums) - one element for each DCT coefficient. It is usually symmetric. Step sizes will be small in the upper left (low frequencies), and large in the upper right (high frequencies); a step size of 1 is the most precise. The quantizer divides the DCT coefficient by its corresponding quantum, then rounds to the nearest integer. Large quantums drive small coefficients down to zero. The result: many high frequency coefficients become zero, and therefore easier to code. The low frequency coefficients undergo only minor adjustment.

Then use quantization table cofficient to divide each dct value to finish quantization

https://www.sciencedirect.com/topics/computer-science/quantization-matrix



Harry Li, Ph.D.

12-27-2018 DCT For 64 Times Image Reduction

DEEP LEARNING BASED HUMAN BEHAVIOR RECOGNITION IN INDUSTRIAL WORKFLOWS

Preprocessing: convert 704×576 color to gray scale image, then
1. Split each frame into 32 × 32 non-overlapping blocks to compute its DCT transform;
2. Keep the 16 most dominant DCT coefficients; (so the reduction is from 32x32 to 4x4)
3. Then perform inverse DCT to recover 4x4 image block.

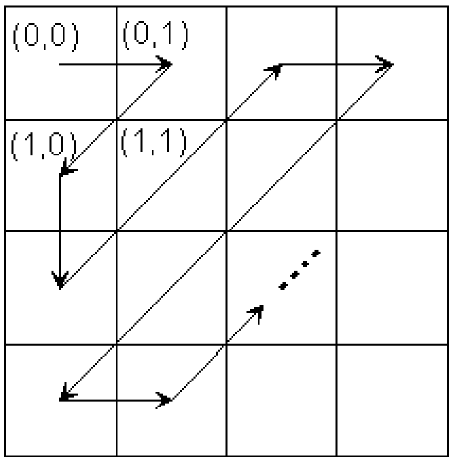
32 × 32 non-overlapping blocks with quantization matrix from (1) luminance, and (2) size extended to 32 x 32 by adding big number 99 as below

16	11	10	16	24	40	51	61
12	12	14	19	26	58	60	55
14	13	16	24	40	57	69	56
14	17	22	29	51	87	80	62
18	22	37	56	68	109	103	77
24	35	55	64	81	104	113	92
49	64	78	87	103	121	120	101
72	92	95	98	112	100	103	99

32

32

16 most dominant DCT coefficients kept in the following manner



4 x 4

Harry Li, Ph.D.

12-27-2018 Confusion Matrix

https://en.wikipedia.org/wiki/Confusion_matrix

A confusion matrix, e.g., an error matrix, is a table for visualization of the performance of an algorithm, typically a supervised learning in machine learning and statistical classification.

Example: If a classification system has been trained to distinguish between cats, dogs and rabbits, a confusion matrix will summarize the results of testing the algorithm. Assuming a sample of 27 animals — 8 cats, 6 dogs, and 13 rabbits, the resulting confusion matrix could look like the table below:

Table 2

		Actual class	
		Cat	**Non-cat**
Predicted class	**Cat**	5 True Positives	2 False Positives
	Non-cat	3 False Negatives	17 True Negatives

Predictive analytics

Table 1

		Actual class			Sub	total
		Cat	**Dog**	**Rabbit**		
Predicted class	**Cat**	**5**	2	0	7	27
	Dog	3	**3**	2	8	
	Rabbit	0	1	**11**	12	
Sub-total		8	6	13		

Total 27 (=8+6+13)

Classification system

Harry Li, Ph.D.

12-26-2018 Pyrdown to Replace DCT for Feature Reduction