People Counting Data for UCSD dataset

Antoni Chan (abchan - AT - ucsd.edu)
Department of Computer Science, City University of Hong Kong

Nuno Vasconcelos (nuno - AT - ece.ucsd.edu)
Dept. of Electrical and Computer Engineering, University of California, San Diego

Feb 28, 2013

This archive contains the feature and count data used in [1, 3] for crowd counting. The counting datasets are based on the two "vidf" and "vidd" scenes in the UCSD pedestrian dataset. The original CVPR dataset contains only 2000 frames from "vidf", while Peds1 and Peds2 each have 4000 frames from "vidf" and "vidd". The three counting datasets are summarized in the following table:

Name	Length	Scene	Videos	File
CVPR [1]	2000 frames	vidf	vidf1_33_000.y to vidf1_33_009.y	cvpr_feats.mat
Peds1 [3]	4000 frames	vidf	vidf1_33_000.y to vidf1_33_019.y	Peds1_feats.mat
Peds2 [3]	4000 frames	vidd	vidd1_33_000.y to vidd1_33_019.y	Peds2_feats.mat

You can find more information in [1, 3], or online at [4, 5, 6]. If you use this data, please cite [1, 3].

1 File Contents

Each MATLAB file contains the feature vectors, counts, and evaluation set (training and test set). Here is a detailed description of the file contents:

dirs{j}	the name for direction j.
fv{j}	[m x n] matrix of feature vectors for direction j. m is the number of features, n is the number of frames. each column is a feature vector.
cnt{j}	[1 x n] matrix of true counts for direction j.
evalsets{n}	set of training/test frames for evaluation. Each cell uses a different number of training frames; the test set is always the same. Some cells contain multiple trials, if the training set is small enough.
evalsets{n}.N	number of frames in the training set.
evalsets{n}.trainset{w}	frame indices for the training set (for trial w).
evalsets{n}.testset{w}	frame indices for the test set (for trial w).
evalsets{n}.trainset2{w}	the "extra" frames not used in the training set, but belong to the training block (for trial w).

The feature vectors are described in the next section. The counts for each direction were obtained using the ground-truth annotations (see README-gt.pdf).

The frame indices start at 1 (MATLAB style). In general, the central part of the video is selected for the training frames, while the remainder is for testing. Using a non-interleaved training

and test sets better demonstrates the generalization of the counting regression function, since the function must extrapolate from the training points. When the training and test set are interleaved, then the regression function is interpolating between training points. Details on the training/test sets are in the table below:

Name	Training Frames	Testing Frames
CVPR [1]	601-1400 (800 frames)	1-600, 1401-2000 (1200 frames)
Peds1 [3]	1401-2600 (1200 frames)	1-1400, 2601-4000 (2800 frames)
Peds2 [3]	1501-2500 (1000 frames)	1-1500, 2501-4000 (3000 frames)

2 Features

The features were extracted from the corresponding motion segments in the video. The region-of-interest and density maps are provided with the ground-truth annotations. The following table lists contents of the feature vector for each dataset. The number is the dimension in the feature vector for that dataset.

	feature description	**CVPR**	**Peds1**	**Peds2**
segment	segment area	1	1	1
	segment perimeter	2	2	2
	perimeter orientation histogram (90 degrees)	3	3	3
	perimeter orientation histogram (120 degrees)	4	4	4
	perimeter orientation histogram (150 degrees)	5	5	5
	perimeter orientation histogram (0 degrees)	6	6	6
	perimeter orientation histogram (30 degrees)	7	7	7
	perimeter orientation histogram (60 degrees)	8	8	8
	perimeter-area ratio	9	9	9
edge	internal edge length	10	10	10
	internal edge orientation histogram (90 degrees)	11	11	11
	internal edge orientation histogram (120 degrees)	12	12	12
	internal edge orientation histogram (150 degrees)	13	13	13
	internal edge orientation histogram (0 degrees)	14	14	14
	internal edge orientation histogram (30 degrees)	15	15	15
	internal edge orientation histogram (60 degrees)	16	16	16
	Minkowski dimension of internal edges	17	17	17
texture	GLCM energy (0 degrees)	18	18	
	GLCM homogeneity (0 degrees)	19	19	
	GLCM entropy (0 degrees)	20	20	
	GLCM energy (45 degrees)	21	21	
	GLCM homogeneity (45 degrees)	22	22	
	GLCM entropy (45 degrees)	23	23	
	GLCM energy (90 degrees)	24	24	
	GLCM homogeneity (90 degrees)	25	25	
	GLCM entropy (90 degrees)	26	26	
	GLCM energy (135 degrees)	27	27	
	GLCM homogeneity (135 degrees)	28	28	
	GLCM entropy (135 degrees)	29	29	
blob	Blob count		30	18

It is advised to normalize each dimension of the feature vector to have zero mean and unit variance. The normalization parameters should be calculated from the training set only.

Note: The features in the CVPR dataset are a little different than those in Peds1 dataset. The stability of the feature extraction code was improved for Peds1. The ground-truth counts for the CVPR dataset are also a little different than Peds1; the CVPR counts include a few more partial people who are leaving the scene. The data provided in `cvpr_feats.mat` was the original data used in [1]. (The updated features and counts for the CVPR dataset are in `cvpr_feats_new.mat`)

3 Example Experiments

The archive includes a MATLAB script `run_experiment.m` for running some experiments from [1, 3]. Some utility functions for are also provided. The experiment code uses the GPML toolbox [7] for learning the GP regression function (results here use version 3.2). Counting results on the various datasets are summarized below. The "Scene" column shows the counting error for all the people (ignoring direction).

	Away (r)		Towards (l)		Scene (t)	
	MSE	|error|	MSE	|error|	MSE	|error|
GPR linear kernel	3.2604	1.4354	2.6918	1.2782	3.6536	1.4893
GPR RBF+RBF kernel	2.9696	1.4075	2.0289	1.0925	3.7868	1.5511
BPR RBF+RBF kernel	2.441	1.210	1.996	1.124	2.975	1.320

Table 1: Counting results on Peds1 (`Peds1_feats.mat`) from [3]

	Right-slow (rs)		Left-slow (ls)		Right-fast (rf)		Left-fast (lf)		Scene (t)	
	MSE	|error|	MSE	|error|	MSE	|error|	MSE	|error|	MSE	|error|
GPR linear kernel	0.6857	0.4850	0.4760	0.4160	0.0093	0.0093	0.0037	0.0037	0.9903	0.6710

Table 2: Counting results on Peds2 (`Peds2_feats.mat`) from [3]. The results here are slightly better than reported in [3] because the code uses a better method to handle redundant features.

	Away (r)		Towards (l)		Scene (t)	
	MSE	|error|	MSE	|error|	MSE	|error|
GPR linear+RBF kernel	4.1808	1.6208	1.2908	0.8692	6.0567	1.9450

Table 3: Counting results on CVPR (`cvpr_feats.mat`) from [1].

	Away (r)		Towards (l)		Scene (t)	
	MSE	|error|	MSE	|error|	MSE	|error|
GPR linear+RBF kernel	3.7950	1.5650	1.3108	0.8575	5.3642	1.7942

Table 4: Counting results on updated CVPR (`cvpr_feats_new.mat`).

4 Acknowledgments

The collection of this database was funded by US National Science Foundation Award IIS-0534985 and CCF-0830535, NSF IGERT Award DGE-0333451, and by the Research Grants Council of Hong

Kong Special Administrative Region of China under grant CityU 110610 (9041552). Part of the ground-truth was annotated by Jeffrey Cuenco and Zhang-Sheng John Liang.

5 History

- 2013/02/28 - initial version

References

[1] A. B. Chan, Z. S. J. Liang, and N. Vasconcelos, "Privacy Preserving Crowd Monitoring: Counting People without People Models or Tracking," In *IEEE Conference on Computer Vision and Pattern Recognition*, June 2008.

[2] A. B. Chan and N. Vasconcelos, "Bayesian Poisson Regression for Crowd Counting," In *IEEE Intl Conf. on Computer Vision*, Kyoto, Sept 2009.

[3] A. B. Chan and N. Vasconcelos, "Counting People with Low-Level Features and Bayesian Regression," *IEEE Trans. on Image Processing*, vol. 21(4), pp. 2160-77, April 2012.

[4] http://www.svcl.ucsd.edu/projects/crowds

[5] http://www.svcl.ucsd.edu/projects/peoplecnt

[6] http://visal.cs.cityu.edu.hk/research/peoplecnt/

[7] http://www.gaussianprocess.org/gpml/code/