A sparse coding framework for gaze prediction in egocentric video

Yujie Li[1], Atsunori Kanemura[1,2], Hideki Asoh[1],
Taiki Miyanishi[2], Motoaki Kawanabe[2]

[1] National Institute of Advanced Industrial Science and Technology (AIST), Tsukuba, Japan

[2] Advanced Telecommunications Research Institute International (ATR), Kyoto, Japan

Egocentric (aka first-person) videos

- **<u>Egocentric videos aka first-person videos (FPVs)</u>**

  Captured by a head-mount camera

  Different from the usual third-person videos (TPVs)

  - Unstable due to head motion

FPVs

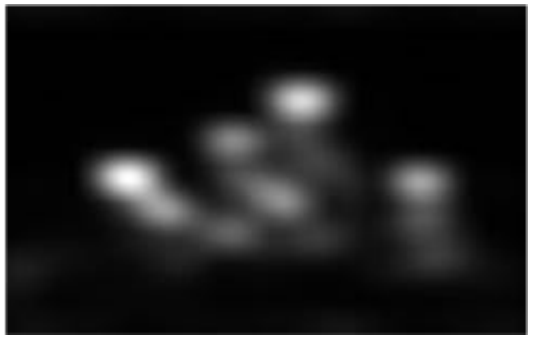
- Background shifts
- Blurs due to head motion
- Only parts of objects

TPV

- Stable background
- With less noise

Gaze detection

- Human gaze
  - Scan the scene by leaping between salient regions
- Saliency detection
  - Has dealt with TPVs
  - Existing methods
    - Itti et al. (*TPAMI* 1998): The classic saliency detection method based on feature integration (ITTI)
    - Harel et al. (*NIPS* 2007): Graph-based visual saliency (GBVS).
    - Li et al. (*CVPR* 2015): Weighted sparse coding framework based on l_1 norm (WSCF).



- We propose a gaze prediction, saliency detection algorithm that works well on FPVs

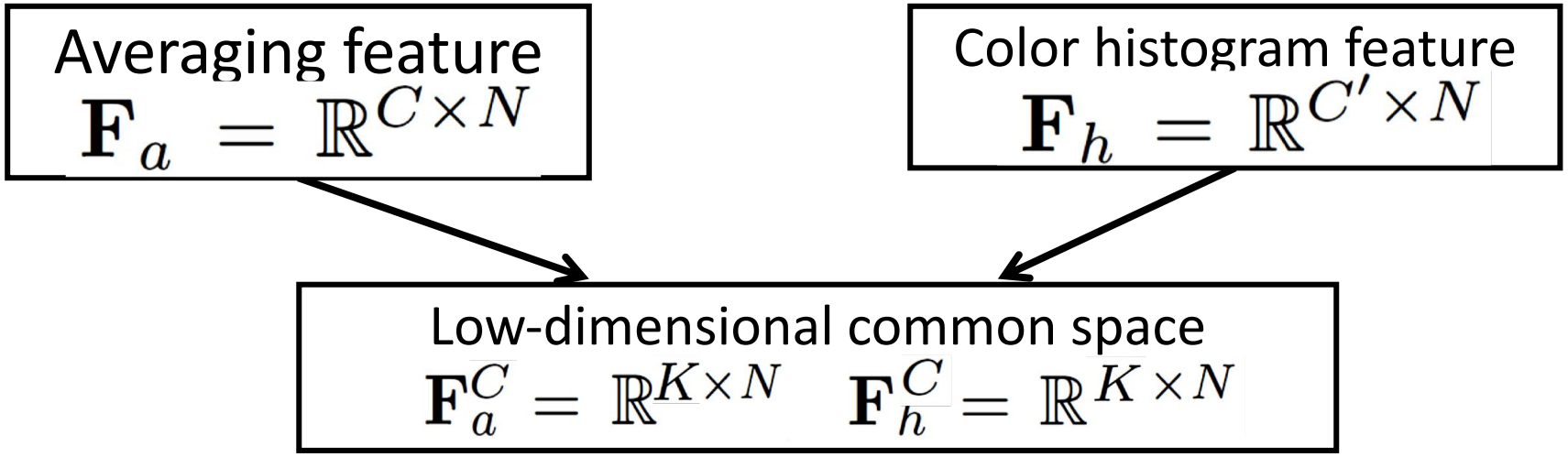
Proposed framework

Simplified procedure

1. Divide an image into superpixels
2. Extract a feature vector for each superpixel
   - Combine intensity averages vector $\boldsymbol{f}_{\mathrm{a}}$ and color histogram vector $\boldsymbol{f}_{\mathrm{h}}$
3. Sparsely reconstruct the feature vector by a saliency dictionary, and evaluate the reconstruction error
   - Saliency dictionary collects features for salient superpixels; it is iteratively updated
4. Identify superpixels whose reconstruction error is smaller than a threshold as salient regions

Feature extraction

- Combine two features by canonical correlation analysis (CCA)
  - Use coupled RGB and Lab color spaces as color descriptors.
  - Generate two feature vectors for all superpixels: An averaged feature vector and a color histogram feature vector.
  - Perform CCA to extract common features



Sparse modeling

- Gaze prediction based on sparse coding (GPSC)
  - Monitor the reconstruction errors of sparse coding with a saliency dictionary.
  - Sparse reconstruction error for superpixel r

$$\epsilon_r^\star = \|\mathbf{f}_r - \mathbf{D}\mathbf{h}_r^\star\|_2^2,$$

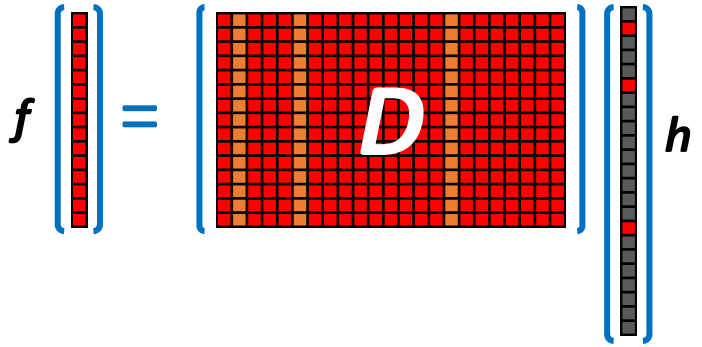
$$\mathbf{h}_r^\star = \underset{\mathbf{h}}{\operatorname{argmin}} \|\mathbf{f}_r - \mathbf{D}\mathbf{h}\|_2^2 \quad \text{s.t. } \|\mathbf{h}\|_0 \leq l$$

  - The saliency value for superpixel r

$$Sal^*(\epsilon_r^\star) = \exp(-\epsilon_r^\star)$$

$$Sal(r) = Sal^+(r) \cdot Sal^*(\epsilon_r^\star),$$

    - $Sal^+(r)$ is a center-bias prior



Dictionary construction

- Initial dictionary
  - Collect superpixels that are distinct from neighboring superpixels (kind of clustering)
- Dictionary updates
  - After calculating saliency values, collect superpixels whose saliencies are larger than a threshold

Experiments

- Dataset: GTEA Gaze (Fathi et al. *ECCV* 2012)
  - Easy cooking activity captured with Tobii eye-tracking glasses
  - True gaze information is available
  - Consisting of 30 actions

Reaching to milk

Taking bread

Putting jam

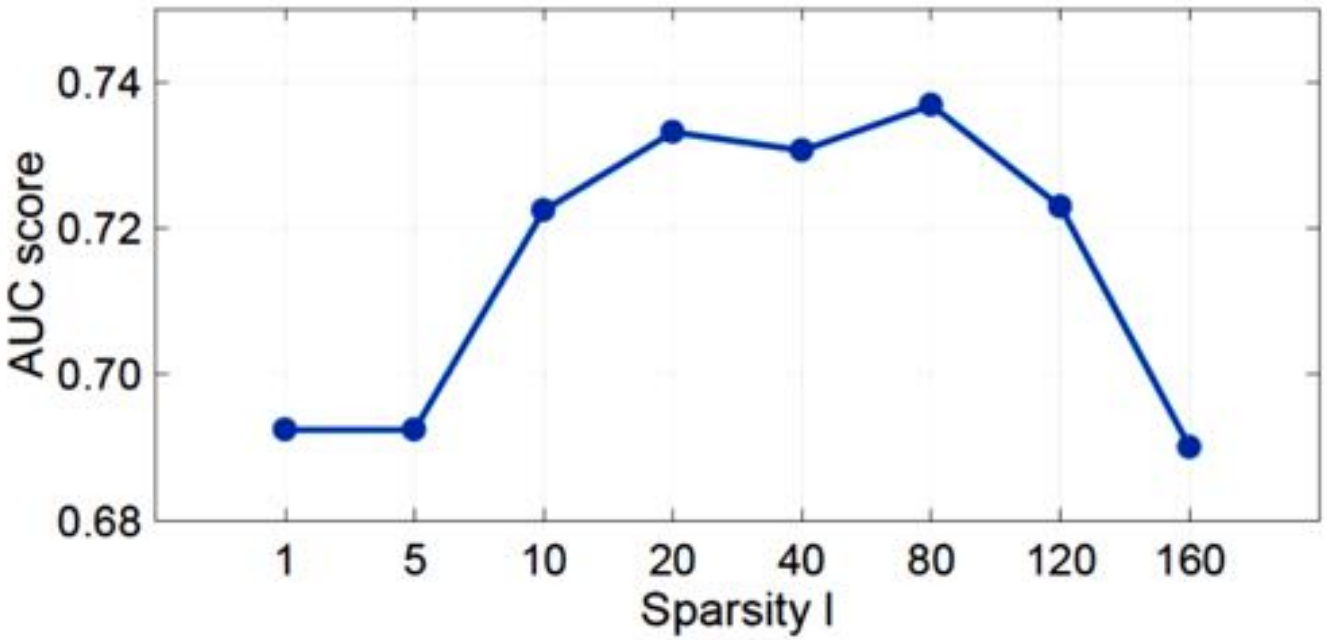
- Evaluation:
  - Receiver operating characteristic (ROC) curve
  - Area under curve (AUC)

Sparsity control

- Degree of sparsity
- Balances the tradeoff
- Less sparse
  - Too few dictionary atoms
  - Not much variety
- More sparse
  - Too many atoms
  - Non-salient regions can be reconstructed well
- We use $l = 50$

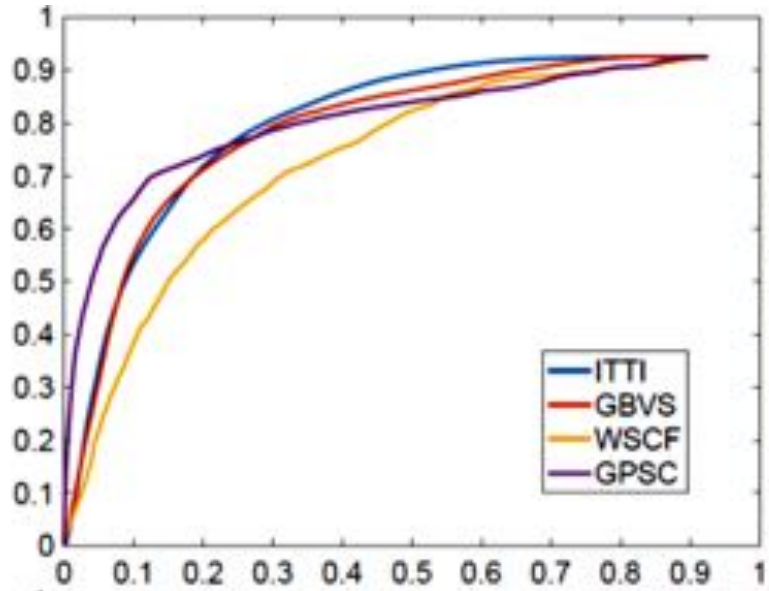
$$\epsilon_r^\star = \|\mathbf{f}_r - \mathbf{D}\mathbf{h}_r^\star\|_2^2,$$

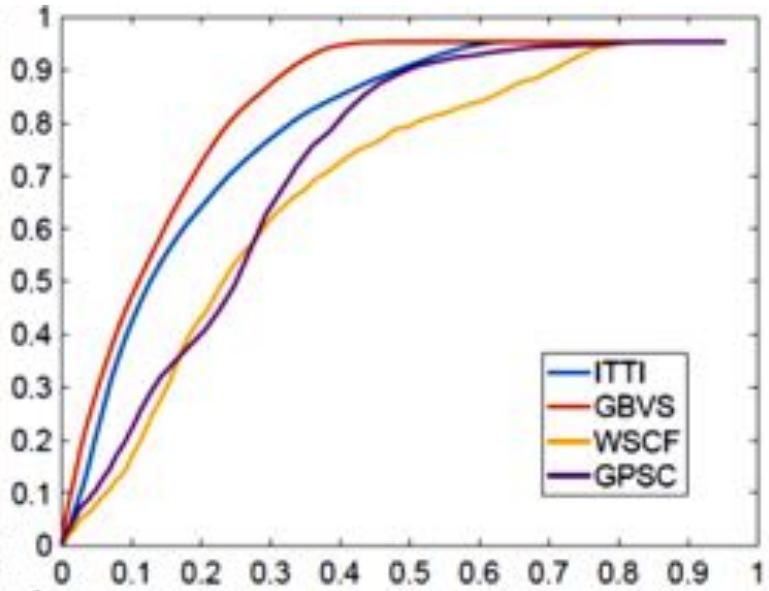
$$\mathbf{h}_r^\star = \underset{\mathbf{h}}{\operatorname{argmin}} \|\mathbf{f}_r - \mathbf{D}\mathbf{h}\|_2^2 \quad \text{s.t. } \|\mathbf{h}\|_0 \leq l$$



ROC curves for different actions

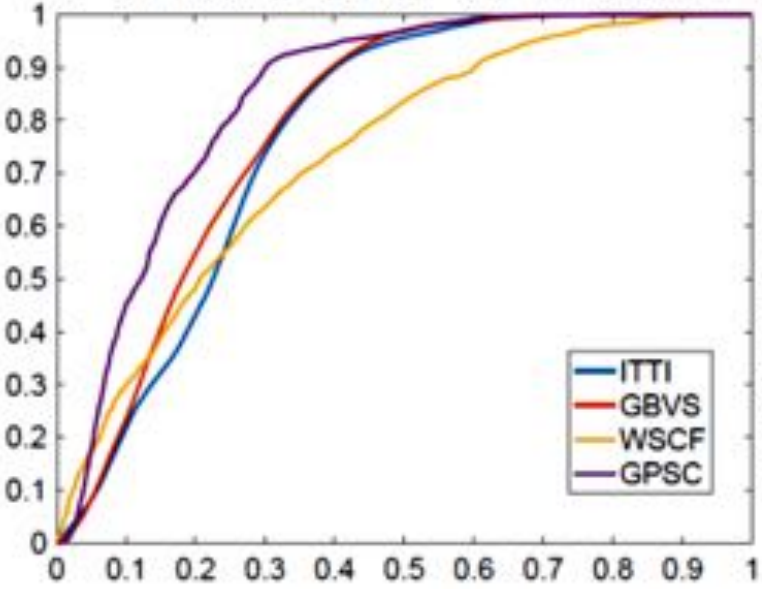
Take bread

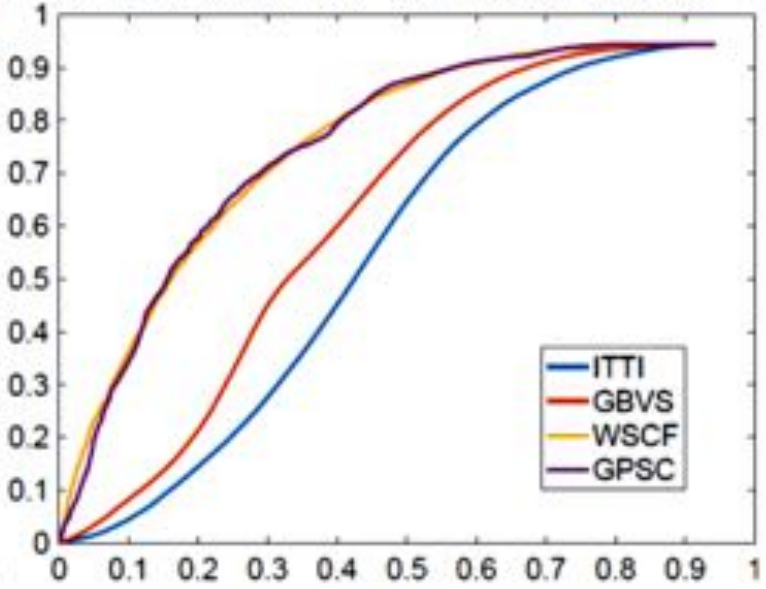




Take plate bowl

Take knife



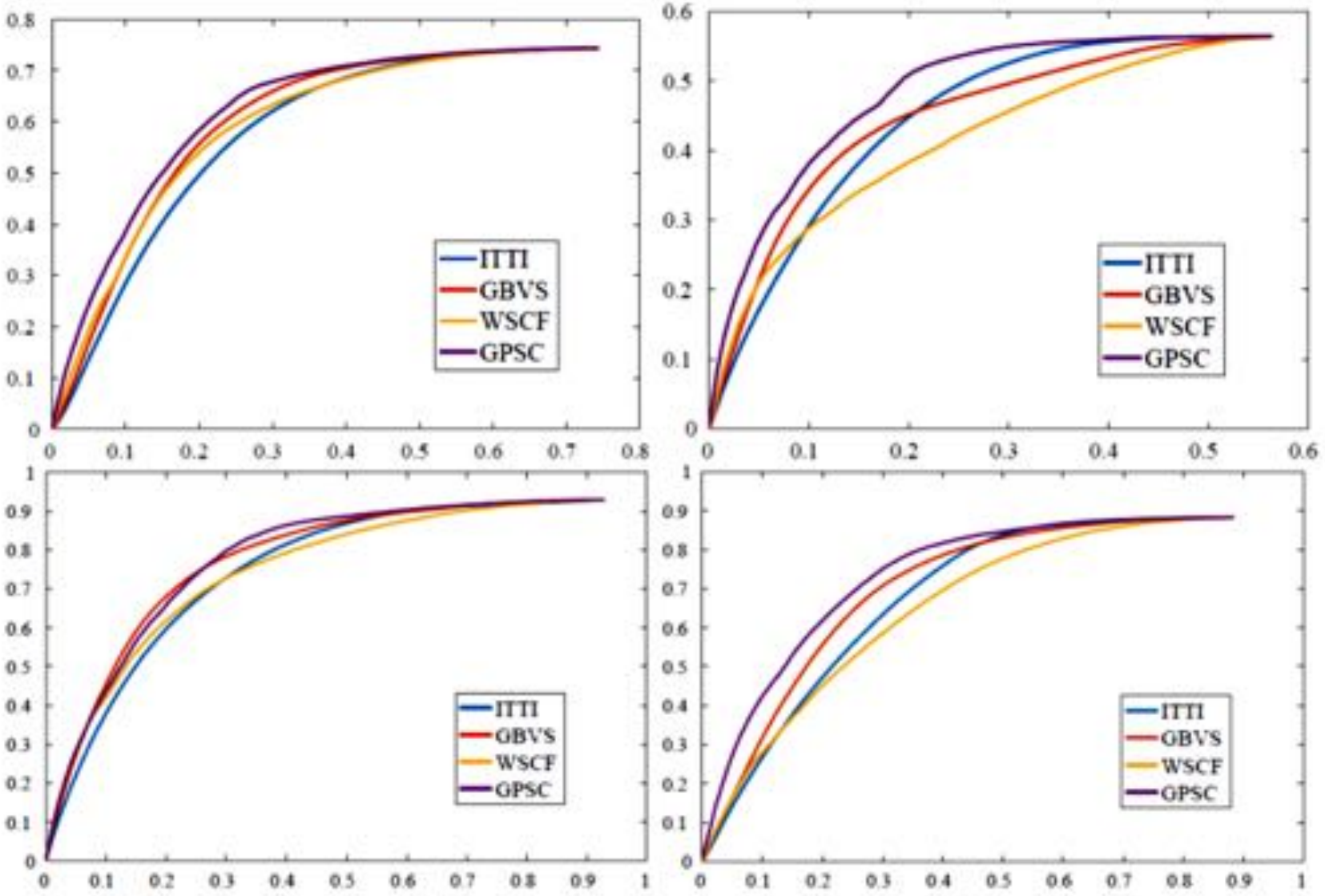


Take bread

AUC scores for different actions

No.	Session name	ITTI	GBVS	WSCF	GPSC
1	take bread	0.733	0.722	0.662	**0.737**
2	take PlateBowl	0.747	**0.789**	0.644	0.682
3	take knife	0.773	0.791	0.737	**0.845**
4	take bread	0.503	0.568	0.701	**0.701**
5	take peanut	0.842	0.810	0.772	**0.859**
6	open peanut	0.630	0.678	**0.830**	0.781
7	scoop peanut	0.554	0.590	0.652	**0.662**
8	spread peanut	0.579	**0.702**	0.590	0.612
9	scoop peanut	0.714	0.769	0.683	**0.795**
10	spread peanut	0.568	**0.632**	0.568	0.552
11	close peanut	0.520	0.501	**0.640**	0.631
12	put peanut	0.621	**0.729**	0.686	0.715
13	take jam	0.775	0.718	0.684	**0.792**
14	open jam	0.866	0.855	0.771	**0.894**
15	spread peanut	0.661	0.752	0.594	**0.778**
16	scoop jam	0.891	**0.906**	0.765	0.898
17	scoop jam	0.882	0.893	0.766	**0.906**
18	close jam	0.870	**0.896**	0.765	0.884
19	put jam	**0.735**	0.666	0.550	0.560
20	spread jam	0.857	0.869	0.758	**0.872**
21	sandwich bread	0.622	0.546	0.551	**0.661**
22	take PlateBowl	0.808	**0.833**	0.745	0.773
23	take cereal	0.736	0.726	0.744	**0.865**
24	pour cereal	0.790	0.867	0.800	**0.871**
25	put cereal	0.671	0.698	**0.714**	0.674
26	take milk	0.593	**0.617**	0.513	0.593
27	open milk	0.855	**0.864**	0.805	0.833
28	pour milk	0.852	**0.879**	0.744	0.873
29	close milk	0.725	**0.761**	0.690	0.715
30	put milk	0.727	0.746	0.681	**0.749**
	mean	0.723	0.746	0.693	**0.759**

ROC curves for different people



AUC scores for different people

No.	002	003	005	006	007	008	010	012
ITTI	0.606	0.418	0.099	0.466	0.664	0.425	0.276	0.426
GBVS	0.598	0.428	0.086	**0.549**	0.697	0.399	0.276	0.444
WSCF	0.571	0.397	0.085	0.420	0.630	0.325	0.253	0.439
GPSC	**0.641**	**0.432**	**0.113**	0.503	**0.714**	**0.444**	**0.297**	**0.458**

013	014	016	017	018	020	021	022	Ave.
0.250	0.690	0.592	0.555	0.581	0.350	0.680	0.513	0.474
0.251	0.717	0.614	0.583	**0.604**	0.371	**0.715**	**0.545**	0.492
0.233	0.691	0.569	0.572	0.579	0.297	0.603	0.439	0.444
0.271	**0.719**	**0.644**	**0.596**	0.603	**0.379**	0.695	0.538	**0.503**

Conclusion

- Gaze prediction on FPVs
- GPSC: CCA-projected features, l_0 norm
- Achieved good performance overall, but not always

- A good ground to build more, e.g. multi-sensor integration