MULTIMODAL SPEAKER LOCALIZATION FROM OMNIDIRECTIONAL VIDEOS

Pascal Reuse, Mihai Gurban*, Ivar Austvoll+ and Jean-Philippe Thiran**

*Signal Processing Laboratory 5,
Ecole Polytechnique Fédérale de Lausanne
Lausanne, Switzerland
email: {pascal.reuse,mihai.gurban,jp.thiran}@epfl.ch
web: splabswww.epfl.ch

+Department of Electrical and Computer Engineering,
University of Stavanger
Stavanger, Norway
email: ivar.austvoll@uis.no
web: www.uis.no

ABSTRACT

The use of omnidirectional cameras for videoconferencing promises to simplify the hardware setup necessary for large groups of participants. We investigate the use of a multimodal speaker detection algorithm on audio-visual sequences captured with such a camera, in particular, an algorithm that uses the audio energy together with the optical flow. We analyze several types of optical flow methods to determine the one which is appropriate to the omnidirectional context.

1. INTRODUCTION

Videoconferencing systems are becoming more and more popular for business and academic communication, due to their ease of use and the ubiquity of internet. For one or two persons et each end, the hardware setup is quite simple, a camera and a microphone. However, when more people are participating, the field of view can become quite crowded and an automatic system to detect the speaker becomes useful, either to move the camera to the person speaking or to switch views between several cameras. But the use of a moving camera or of multiple cameras makes the hardware system more complicated. A setup with just one omnidirectional camera is simpler and still allows changing the view and focusing on the current speaker automatically. This is our target scenario.

We investigate the use of a multimodal speaker localization algorithm on omnidirectional sequences. In particular, as our algorithm uses optical flow for the video processing part, we analyze the influence of the type of optical flow extraction method on the speaker localization, with the specific constraints of omnidirectional video.

This article continues and expands our previous work presented in [1]. Our first contribution here is the adaptation of our multimodal speaker localization technique to the use on omnidirectional videos. Our second contribution is the analysis of several optical flow extraction methods, showing their particularities in the case of visual speech.

The structure of the paper is as follows. First, we give the context of our work, presenting a brief overview of the state of the art in speaker localization, optical flow computation and omnidirectional image processing. Second, we compare several optical flow methods for their use in speech analysis. We aim to find the optical flow method which is best suited to capture the motion of the mouth, in the particular context of omnidirectional images. Finally, we present our results for omnidirectional speaker localization, on sequences acquired in our laboratory.

2. THE CONTEXT

2.1 Speaker localization

It is possible to perform speaker localization only from the audio modality, using a microphone array, but we are aiming for a simple hardware setup, so only a simple microphone will be used. Previous approaches to audio-visual speaker localization either assume the local Gaussianity of the data, as in [2] [3], or rely on complex and computing-intensive operations at test time to detect correlation between the audio and the video [4] [5] [6].

We will use our speaker localization method detailed in [1], which uses the joint audio-visual probability density to find the most likely locations of the current speaker's mouth. This multimodal speaker localization algorithm has several advantages. First, the use of a training procedure ensures that the number of operations performed while testing is reduced, making possible a real-time implementation. However, optical flow extraction is still very heavy computationally, and there are possible optimizations which will be discussed in the next section.

Another advantage of our approach is that, in contrast to methods that consider the audio and video features of speech to have a Gaussian joint probability density, we can model any kind of probability density. The Gaussian mixture model that we use is an universal approximator of densities, provided that enough Gaussians are considered.

Finally, in our case, no face tracker needs to be used, as testing is done on the entire image, not only the face or mouth region. An extracted mouth region is required, but only in the training step.

For more details on the state of the art in speaker localization, and on our localization method, we direct the reader to our paper, [1].

In the next section, we will give a short overview of optical flow methods, and in particular of the algorithms used in this work.

2.2 Optical flow algorithms

The optical flow is the apparent motion of objects in an image. There is usually a difference between this *apparent* motion and the *real* motion which is the projection of the object's 3D motion on the 2D image plane. A simple example that can be given is a uniform sphere which is illuminated by a light source which is rotating around it. The optical flow seen on the surface of the sphere would suggest that it is moving, although in reality it is static.

The optical flow (OF) is represented with a field of motion vectors, one for each pixel in the image. The OF, as de-

fined by the Brightness Constraint Equation, is an ill-posed problem. A simple model of the imaging system is a pinhole camera. If we assume that the illumination is given by a point source at infinity and a single moving object is considered, and that the reflectance of the object is Lambertian and that there is no photometric distortion, then the difference between the real velocity in the image plane and the OF can be shown to be [7] $|\Delta \mathbf{v}| = \rho |\mathbf{I}^T \boldsymbol{\omega} \times \mathbf{n}| / \|\nabla E\|$, where ρ is the surface *albedo*, $\mathbf{I}$ is the illumination, $\boldsymbol{\omega}$ is the angular velocity, $\mathbf{n}$ is the surface normal and ∇E is the spatial gradient of the image gray level. This shows that usually $|\Delta \mathbf{v}| \neq 0$. We can summarize the result as follows. The error is small (or zero) when

- the spatial gradient, $\|\nabla E\|$, is large, or
- the 3D velocity is a translation (in any direction), i.e. $\boldsymbol{\omega} = \mathbf{0}$, or
- the direction of illumination, $\mathbf{n}$, is parallel to the angular velocity, i.e. $\boldsymbol{\omega} \times \mathbf{n} = \mathbf{0}$.

In addition we give the following remark. The resulting OF depend on the direction of the 3D translation. In the case of a translation towards or away from the camera we get a diverging flow field (expanding or contracting respectively). If the translation is parallel to the image sensor the flow is also parallel with a vector length decreasing with increasing distance to the camera.

In this application, with sufficient lighting, rotations can be ignored and only a translational velocity field parallel to the viewing direction is assumed. The distance to the speakers are more or less constant such that the OF vectors is a sufficient good description of motion of facial features.

There are several categories of algorithms which can be used for the computation of the optical flow [8]:

- Differential methods - which use spatiotemporal derivatives of image intensities.
- Correlation-based methods - which use matching of regions in the image.
- Spatio-temporal directional filtering - which use orientation sensitive filters.

We included several methods in our evaluation, aiming to find one that is well-suited to our problem. The Horn and Schunck [9] method is a differential method, which imposes a velocity smoothness constraint in order to find the optical flow vectors. The Lukas and Kanade [10] method solves the optical flow constraint equations for groups of adjacent pixels, assuming they move with more or less the same velocity. The differential methods work well for small velocity vectors. In the case of large velocities a multiresolution approach can be used.

We also tested a block matching method [11], which may be more appropriate than differentiation in the case of poor signal to noise ratios and low framerates, since for low framerates the velocity vectors are usually large. Typically, matching amounts to maximizing a similarity measure [8].

We will not go into more details for the methods mentioned previously, however we will briefly present the estimation of optical flow with directional filters [12] since we think it could be a better fit with omnidirectional images.

The method consists in the application of filters at different scales of the image, as shown in fig. 1. The local image model used by this method is the modulated complex exponential. Input images (in the video sequence) are first given

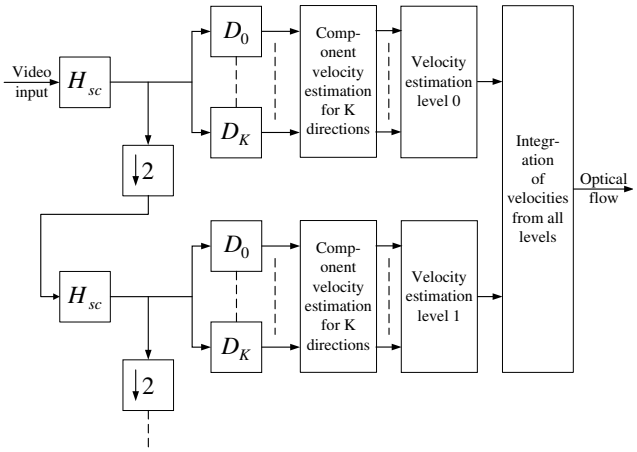

Figure 1: An overview of the directional filters algorithm, taken from [12].

to a 2D low-pass spatial filtering (H_{sc}) that bounds the maximal spatial frequency to prevent aliasing in the next steps. Then, images are filtered using directional filters (D_0 ... D_k), described in detail in [13]. These filters are one-sided (complex) bandpass filters that correspond to bandpass-filtering followed by a Hilbert transform. For each direction a 2D complex signal is extracted with spatial coordinate, s in the given direction and time coordinate, t. This signal is locally modeled as $f(s,t) = \exp[j\phi(s,t)]$. The component velocity in this direction is found by estimation of the angle of the local structure. In [13] the phase signal is used and the direction computed by a set of quadrature filters. In [12] the gradient (equivalent to the instantaneous frequency) is estimated by an AM-FM demodulation scheme [14] and the structure tensor computed by the outer product. The component velocity is found by *tan* to the angle of the eigenvector belonging to the smallest eigenvalue. The full OF vector is estimated by combining a set of at least two component velocities in different directions (for more than two a least square solution is used). A range of spatial frequencies can be covered by using several levels. Lower spatial frequencies make it possible to estimate larger velocity vectors. At level 2 the speed range is from zero to approximately 5 pixels.

In this application only one direction is needed and only one scale level is sufficient. This reduces the computational complexity considerably. For the speakers in a distance of one meter from the camera the velocity is most of the time in the range of less than 5 pixels such that we can use level 2 and a down sampling by 2.

The low-pass spatial filter is also necessary for the other methods in order to remove noise.

2.3 Omnidirectional images

Conventional imaging systems have a limited field of view, which can be expanded either through the use of multiple cameras, or through the addition of a moving mount. Since these are expensive, an alternative is to use mirrors together with the lenses to expand the field of view. A catadioptric sensor uses a combination of a curved mirror and a lens to form a projection on a camera's sensor. The types of mirrors used can be spherical, hyperbolical and parabolical [15].

Omnidirectional images are taken with a catadioptric

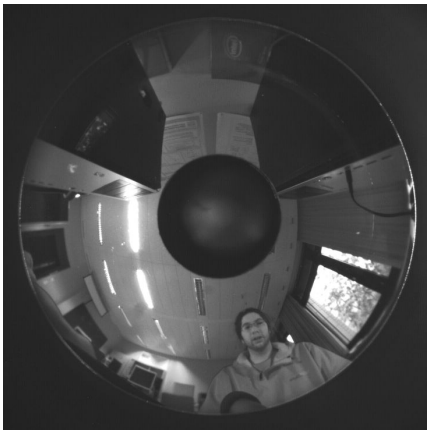

Figure 2: An example of an omnidirectional image taken with our camera.

setup with a 360° field of view. This is the setup that we will use in our experiments. Our aim is to simplify the hardware configuration used for videoconferencing, allowing larger number of participants with a single visual camera.

The geometrical properties of the omnidirectional image, which is obviously distorted, are well-studied. It is possible to reconstruct the whole visual information on a sphere, around a perfect point-like observer standing at the focus of the mirror [15]. However, this transformation is expensive and not necessary in our application's context. We aim to find the movement in the image which corresponds to an active speaker, and for this only the vertical motion is required [1]. This vertical motion becomes radial in omnidirectional image, so in the end what is required is an optical flow extraction method which can determine efficiently the motion vectors distributed radially around the center of the image.

In this case, the directional filters method seems ideally suited for this task. Indeed, instead of computing the optical flow in all directions throughout the image, only filters adjusted to the particular radial direction required could be used, greatly reducing the processing time. However, our findings show that the frame rate attained for the spherical images is insufficient to obtain good results with this optical flow method. The next section shows a comparison between the different optical flow algorithms on two types of sequences, planar and omnidirectional.

3. OPTICAL FLOW EVALUATION FOR SPEECH

First, we want to evaluate the performance of the directional optical flow method for planar sequences, taken from the CUAVE database [16]. The result is shown in fig. 3. The sequences are at 60 deinterlaced frames per second (fps), and the resolution is 720x480. As can be seen, in these conditions the directional filters method performs better than Horn and Schunck [9], as the motion is smoother. The optical flow is computed around the speaker's mouth. The third part of the figure shows the space-time slice on this region.

For omnidirectional images, the conditions are different, as the high-resolution camera is unable to provide us an equally high fps. The resolution of the images is 1600x1000, but the fps is only 15. Fig. 4 shows the comparison between the Horn and Schunck [9] method, Lukas and Kanade [10] and block matching, together with the space-time slice

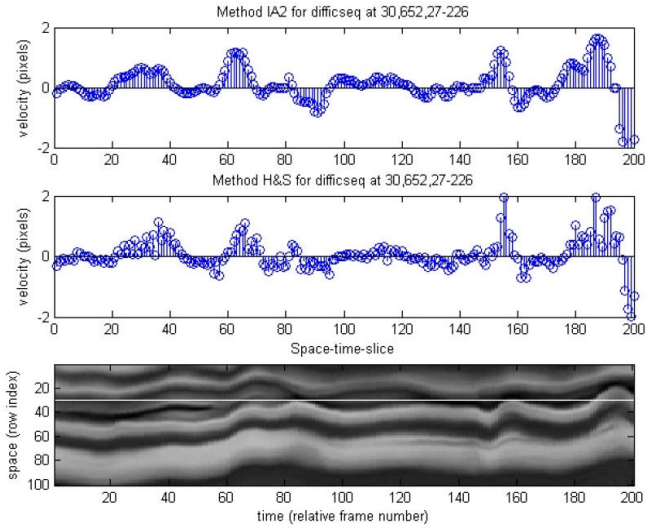

Figure 3: Optical flow comparison at 60 fps. Here the directional filters method is compared to Horn and Schunck.

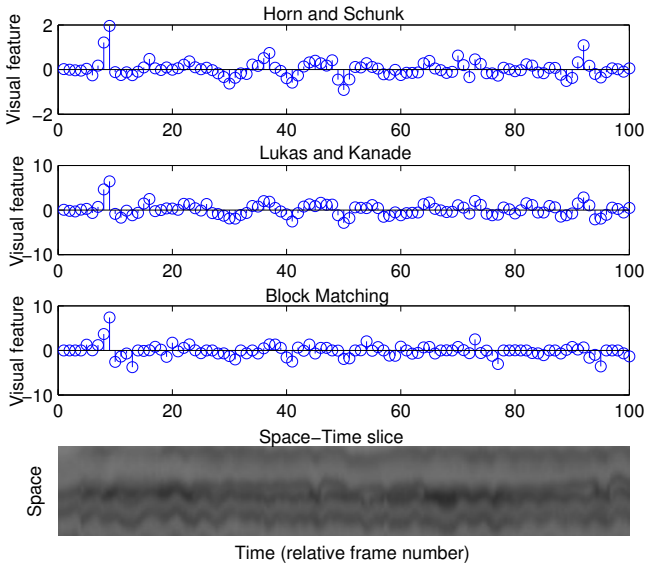

Figure 4: Optical flow comparison at 15 fps.

derived from the directional filters. Unfortunately, the directional filters method is unable to provide us with accurate optical flow vectors at such a low temporal resolution. This may be caused by the fact that this method uses a time window of several frames to compute the motion in the image, making it difficult at this frame rate to accurately follow the motion of the mouth.

In conclusion, the directional filters method, although promising, could not be used for speaker detection in omnidirectional images. It is clear that with a camera able to provide a higher fps this method would work better, but using it with our particular hardware we were unable to obtain good results. From the other methods tested, the Horn and Schunck [9] method was the most accurate on our data, and it was chosen for our following experiments.

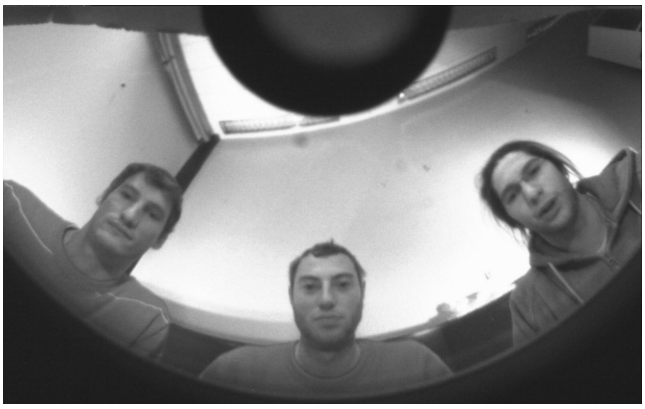

Figure 5: The cropped field of view in our experiments.

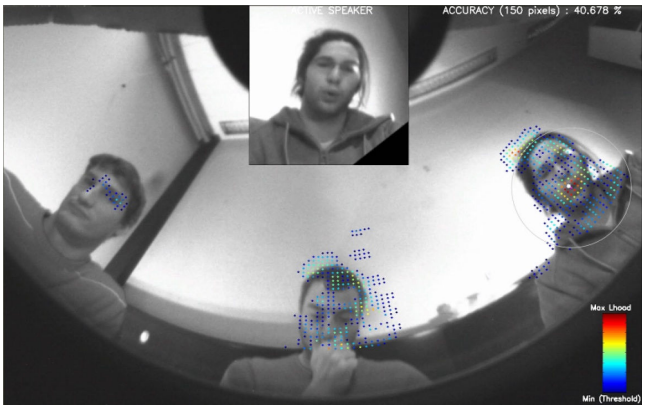

Figure 6: A test frame with superposed likelihoods and a circle of radius 150 around the ground truth location.

4. TRACKING FROM OMNIDIRECTIONAL VIDEOS

We used a high-resolution (3264x2448) camera, but, unfortunately, at this resolution the frame rate was only 3 fps, unusable for analyzing speech. By cropping the image and limiting the field of view to around 130°, as shown in fig. 5, we were able to increase the rate to 15 fps. As shown above, this was enough to estimate optical flow with the H&S method, but not with directional filters. Our findings on the cropped images are also valid on the full 360° field of view, the only limitation here being the hardware.

In our speaker tracking experiments, we used the method detailed in [1], which consists in building a joint audio-visual probability density function (pdf) using Gaussian mixture models, and then using this model to determine the likelihood of finding the speaker's mouth at each position in the test image. The features used are the audio energy and the vertical difference of optical flow vectors, as detailed in the paper.

Since an omnidirectional audio-visual database suited for speaker tracking does not exist, we recorded our own sequences, both for training and for testing. In total, 6 test sequences were recorded, with 3 speakers taking turns saying digits. The length of each sequence was between 1 and 1.5 minutes. The layout is shown in fig. 5. We decided to do all the processing in the plane domain, although computing optical flow directly on the sphere might have been more precise. Our choice is motivated by the fact that the system needs to be fast, and distortions at the level of the speakers' faces were not significant.

For the moment, the optical flow is computed along the vertical and the horizontal in the image plane, and then it is projected along radial dimensions around the center of the field of view, which is at the top of the image. From these motion vectors spatial differences are computed on a radial grid, giving us the visual features.

Our 6 test sequences were recorded with different assumptions on the motion of the speakers:

- Sequences 1 and 2: static speakers, the persons only move their mouths.
- Sequences 3 and 4: natural relaxed stance, heads and bodies move according to what is being said.
- Sequences 5 and 6: realistic motion, including irrelevant movements like chin scratching, even when the person is not speaking.

Aside from the speaker tracking algorithm, we also implemented a very simple silence detector, based on the energy of the audio, to set aside the segments in which nobody was speaking. To obtain quantitative results, we also established a frame-level ground truth, in which the position of the current speaker's mouth was located manually.

The location found by our algorithm was compared to the ground truth position for each frame in the sequence. If the detected position was inside a circle of radius r around the ground truth, we considered the frame a successful match. We used three values for the circle radius, $r = 70$, $r = 100$ and $r = 150$. Fig. 6 shows a test image together with superposed likelihood values, showing the probability of the speaker's mouth being at that location. The ground truth position is also shown, together with a circle of 150 pixels around it. As can be seen from the figure, even a circle of $r = 150$ is enough for a good localization of the current speaker. As typically the head might move in the same rhythm as the speech, sometimes the chin or the front of the speaker are wrongly identified as the mouth, although the algorithm is correctly identifying the speaker.

Fig. 7 shows our results for the 6 sequences, from the simplest to the most difficult. Numerical results are presented in table 1. As can be seen, for the simplest case, when the speakers are still and only the mouth is moving, results are close to 100%. The mouth is correctly located even with a small radius circle around the ground truth. As the sequences become more and more difficult, the accuracy decreases, reaching 40% for the last sequence, with $r = 70$. However, if the radius of the circle is increased, the accuracy increases to 72%, which shows, as mentioned before, that although the mouth itself might not be correctly localized, the speaker is, because his head is moving in synchrony with the speech. This effect can account for the difference between results with $r = 70$ and $r = 150$.

The effect of the silence detection step is not very pronounced in these sequences, since there are very few segments in which all the speakers are silent. The improvement brought by the silence detector is around 2% for all sequences.

Many of the errors found are in fact caused by the incorrect estimation of the optical flow. There are instances where although the mouth is moving, the motion vectors on

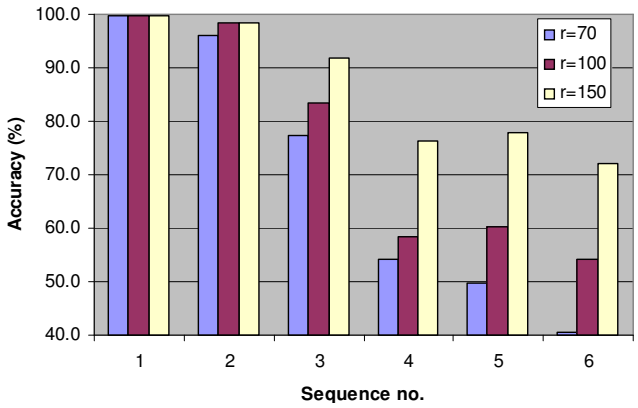

Figure 7: Results for the 6 test sequences, from easiest to most difficult, with three different values for the tolerated radius around the speaker's mouth.

Seq. no.	r=70	r=100	r=150
1	99.8	99.8	99.8
2	96.1	98.5	98.5
3	77.5	83.5	91.8
4	54.2	58.5	76.4
5	49.7	60.3	77.8
6	40.5	54.2	72.0
Avg.	69.6	75.8	86.1

Table 1: Speaker tracking accuracy.

the region of the mouth are quite small, making identification impossible. However, as our method uses the average over a temporal window, some of these errors will disappear because of the averaging. It is clear though that a more precise optical flow method would greatly improve results.

Our average result of 86% correct recognition over the 6 omnidirectional sequences shows that it is possible at this time to use an omnidirectional camera for speaker localization, simplifying the hardware setup necessary for a video-conferencing system.

5. CONCLUSION

Our work had two main aims. The first one was to evaluate if it was possible in practice to use an omnidirectional imaging system for speaker localization, while the second was to evaluate several optical flow algorithms and to find the one which is best suited for this task.

We have shown that our audio-visual speaker tracking algorithm performs quite well with an omnidirectional camera. In spite of the low frame rate, we achieved a good performance in a wide range of conditions, correctly finding the current speaker in the omnidirectional image.

The optical flow evaluation has shown that a higher frame rate is however needed, before better optical flow methods can be used, in particular the directional filters method, which promises to find the required motion vectors with better accuracy and less computation.

Acknowledgements

This work is supported by the Swiss National Science Foundation through the IM2 NCCR.

REFERENCES

[1] M. Gurban and J.Ph. Thiran, "Multimodal speaker localization in a probabilistic framework," in *Proceedings of the 14th European Signal Processing Conference (EUSIPCO)*, 2006.

[2] J. Hershey and J.R. Movellan, "Audio vision: Using audio-visual synchrony to locate sounds," *Neural Information Processing Systems*, pp. 813–819, 1999.

[3] H.J. Nock, G. Iyengar, and C. Neti, "Speaker localisation using audio-visual synchrony: An empirical study," *Proceedings of the International Conference on Image and Video Retrieval*, 2003.

[4] J.W. Fisher III and T. Darrell, "Speaker association with signal-level audiovisual fusion," *IEEE Transactions on Multimedia*, vol. 6, no. 3, pp. 406–413, 2004.

[5] T. Butz and J.P. Thiran, "From error probability to information theoretic (multi-modal) signal processing," *Signal Processing*, , no. 85, pp. 875–902, 2005.

[6] P. Besson, M. Kunt, T. Butz, and J.P. Thiran, "A multimodal approach to extract optimized audio features for speaker detection," *Proceedings of the European Signal Processing Conference (EUSIPCO)*, 2005.

[7] Emanuele Trucco and Alessandro Verri, *Introductory Techniques for 3-D Computer Vision*, Prentice Hall, 1998.

[8] S. S. Beauchemin and J. L. Barron, "Computation of optical flow," in *ACM Computing Surveys*, 1995, vol. 27, pp. 433–467.

[9] B. Horn and B. Schunck, "Determining optical flow," in *Artificial Intelligence*, 1981, vol. 17, pp. 185–204.

[10] B.D. Lucas and T. Kanade, "An iterative image registration technique with an application to stereo vision," *Proceedings of the International Joint Conference on Artificial Intelligence*, pp. 674–679, 1981.

[11] J. Jain and A. Jain, "Displacement measurement and its application in interframe image coding," *Communications, IEEE Transactions on*, vol. 29, no. 12, pp. 1799–1808, Dec 1981.

[12] E. Kristoffersen, I. Austvoll, and K. Engan, "Dense motion field estimation using spatial filtering and quasi eigenfunction approximations," in *ICIP (3)*, 2005, pp. 1268–1271.

[13] I. Austvoll, "Directional filters and a new structure for estimation of optical flow," in *ICIP*, 2000.

[14] J.P. Havlicek, D.S. Harding, and A.C. Bovik, "Discrete quasi-eigenfunction approximation for am-fm image analysis," in *Proceedings of the IEEE int. Conf. on Image Processing*, 1996, pp. 633–636.

[15] C. Geyer and K. Daniilidis, "Catadioptric projective geometry," *Int. J. Comput. Vision*, vol. 45, no. 3, pp. 223–243, 2001.

[16] E.K. Patterson, S. Gurbuz, Z. Tufekci, and J.N. Gowdy, "Moving-talker, speaker-independent feature study and baseline results using the CUAVE multimodal speech corpus," *EURASIP Journal on Applied Signal Processing*, vol. 2002(11), pp. 1189–1201, 2002.