

# A Graph Representation of Filter Networks

Björn Svensson, Mats Andersson, and Hans Knutsson\*

Department of Biomedical Engineering, Medical Informatics  
Center for Medical Image Science and Visualization  
Linköping University, Sweden  
{bjosv, matsa, knutte}@imt.liu.se

**Abstract.** Filter networks, i.e. decomposition of a filter set into a layered structure of sparse subfilters has been proven successful for e.g. efficient convolution using finite impulse response filters. The efficiency is due to the significantly reduced number of multiplications and additions per data sample that is required. The computational gain is dependent on the choice of network structure and the graph representation compactly incorporates the network structure in the design objectives. Consequently the graph representation forms a framework for searching the optimal network structure. It also removes the requirement of a layered structure, at the cost of a less compact representation.

## 1 Introduction

Filter networks for efficient convolution [1] is a technique for designing and implementing sets of multidimensional finite impulse response (FIR) filters with significantly lower complexity compared to standard convolution. Successful design and practical use of filter networks is shown in e.g. [2], where local 3-D structure is extracted from visual data. This paper shows how elementary graph theory can be used to compactly represent a filter network. The graph representation aims to form a framework for future work on a more general strategy for design of filter networks. Several other design techniques fits within this framework and a few samples are given to illustrate the concept.

Filter networks for efficient convolution are single-rate systems and are not to be confused with the widely used multi-rate systems. The purpose of multi-rate system is data compression using signal decimation rather than efficient filtering. Identical filtering can be performed by a single-rate system, using sparse subfilters. This is a classic example of trading memory for speed. Interpolation is not necessary for single-rate systems which increase the computational efficiency, while multi-rate systems require less amount of memory due to signal decimation. Design of multi-rate systems often starts out from the condition of perfect reconstruction, which is not a requirement for general single-rate systems designed for efficient convolution.

---

\* The financial support from The Swedish Agency for Innovation, VINNOVA, The Swedish Foundation for Strategic Research, SSF, and ContextVision AB is gratefully acknowledged.

## 2 FIR Filter Design

The amount of research devoted to the classic problem of FIR filter design indicates its importance as a fundamental operation in signal processing applications. Much interest has been directed towards the design of 1-D equiripple low-pass filters. An equiripple filter is obtained by finding the Chebyshev approximation to the desired frequency response. This problem was solved by McClellan-Parks [3], using the Remez exchange algorithm.

The Remez exchange algorithm is based on the alternation theorem, which is only applicable on 1-D filters. Due to this, focus for design of multidimensional filters mainly turned towards the weighted least mean squares (WLMS) technique [4] and the eigenfilter approach [5]. Both approaches allow constraints or objectives in the spatio-temporal domain, which is not the case for the traditional approach presented by McClellan-Parks.

### 2.1 Least Squares Design

The problem of FIR filter design is to choose the complex coefficients  $\mathbf{c} \in \mathbb{C}^N$  of the discrete impulse response  $\tilde{f}(\boldsymbol{\xi})$ , with the closest fit to, in the general case, a number of desired functions. In this paper two objectives,  $\alpha$  in the frequency domain and  $\beta$  in the spatio-temporal domain are used. Each coefficient  $c$  is associated with a discrete spatio-temporal position  $\boldsymbol{\xi} \in \mathbb{Z}^n$  on a Cartesian grid. The efficiency of a filter, i.e. the number of multiplications and additions per data sample, is determined by the number of nonzero filter coefficients  $N$ , which grows exponentially with the signal dimensionality  $n$ .

A direct implementation yields a frequency response  $\tilde{F}(\mathbf{u})$ , which is linear w.r.t. the nonzero coefficients  $\mathbf{c} = [c_1, c_2, \dots, c_N]^T \in \mathbb{C}^N$  of the impulse response  $\tilde{f}(\boldsymbol{\xi})$  due to the Fourier transform  $\tilde{F}(\mathbf{u}) = \mathcal{F}\{\tilde{f}(\boldsymbol{\xi})\}$  in Eq. 1.

$$\tilde{F}(\mathbf{u}) = \sum_{\mathbb{Z}^n} \tilde{f}(\boldsymbol{\xi}) \exp(-i\boldsymbol{\xi}^T \mathbf{u}) = \sum_{k=1}^N c_k \exp(-i\boldsymbol{\xi}_k^T \mathbf{u}) \tag{1}$$

The frequency objective  $\alpha(\mathbf{c})$  in Eq. 2 describe a WLMS-error between the frequency response  $\tilde{F}(\mathbf{u})$  of  $\tilde{f}(\boldsymbol{\xi})$  and the desired frequency response  $F(\mathbf{u})$ . Similarly the spatio-temporal objective  $\beta$  in Eq. 3 is expressed as the WLMS-error between the impulse response  $\tilde{f}(\boldsymbol{\xi})$  and the desired impulse response  $f(\boldsymbol{\xi})$ .

$$\alpha(\mathbf{c}) = \int_{\mathbb{U}} W(\mathbf{u}) \left| F(\mathbf{u}) - \tilde{F}(\mathbf{u}) \right|^2 d\mathbf{u}, \quad \mathbb{U} = \{\mathbf{u} \in \mathbb{R}^n : |u_i| \leq \pi\} \tag{2}$$

$$\beta(\mathbf{c}) = \sum_{\mathbb{Z}^n} w(\boldsymbol{\xi}) \left| f(\boldsymbol{\xi}) - \tilde{f}(\boldsymbol{\xi}) \right|^2 \tag{3}$$

The optimal impulse response  $\tilde{f}^*$  with nonzero coefficients  $\mathbf{c}^*$  is here obtained by simultaneously minimizing  $\alpha(\mathbf{c})$  and  $\beta(\mathbf{c})$  in Eq. 4.

$$\mathbf{c}^* = \arg \min_{\mathbf{c} \in \mathbb{C}^N} \alpha(\mathbf{c}) + \beta(\mathbf{c}) \tag{4}$$

The Chebyshev approximation corresponds to having a frequency objective  $\alpha(\mathbf{c})$ , where the  $l_2$ -norm is replaced by the  $l_\infty$ -norm, while ignoring the spatio-temporal objective. For 2-D filters [6] these equiripple designs can still be achieved by in an iterative manner updating the weighting functions. The equiripple property is however rarely desired in image processing applications.

## 2.2 Multispace Design and Weighting

Filtering, using FIR filters is of course application dependent. The choice of spaces and associated weighting functions should therefore naturally be based on a priori information. As opposed to most design methods, this is here incorporated in the filter design. In this paper the design objectives are restricted to the Fourier space and the spatio-temporal space, but the least squares approach can easily be extended to arbitrary number of objectives in multiple design spaces [7].

In the Fourier space, the weighting function preferably favors a close fit to the desired frequency response for the signal frequencies most critical for the application in mind. Consequently, the errors are distributed among the less critical frequencies. A natural approach is to use a weighting function, which favors a close fit to the most common frequencies, i.e. the expected signal and noise spectra. The spatio-temporal objective can be used to favor locality, an important property to prevent the filters for mixing up different events present in the data.

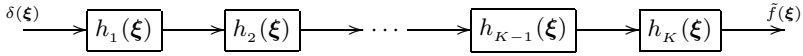
## 3 Design of Cascaded Filters

The idea of decomposing filters into cascaded sparse subfilters  $h_k(\boldsymbol{\xi})$  (see Fig. 1) for computationally efficiency is far from new and an overview of early work on this topic is given in [8]. Early research mostly concerns narrowband 1-D filters, since sharp transition bands are hard to implement efficiently using standard FIR filters. Cascaded filters for other purposes have not been very popular, since it in general offers no improvement of computational efficiency for 1-D filters.

The approaches are divided into those that reuse the same subfilter (see e.g. [9]) and those who use different subfilters. The two single-rate approaches with most impact, the frequency-response masking technique [10] and interpolated FIR filters are closely related. The frequency response masking technique is actually a small filter network rather than cascaded subfilters.

Interpolated FIR filters, known as IFIR, was introduced in [11]. The basic idea is to use one sparse subfilter followed by a nonsparse subfilter acting as an interpolator in the spatio-temporal domain. By recursively applying this strategy, the interpolator itself can be divided into a new pair of subfilters, a sparse one and an interpolator. In this way a filter sequence with length larger than 2 is obtained.

As opposed to design of 1-D filters, cascaded multidimensional filters generally improve the computational efficiency. Just consider convolution between two filters of spatial size  $N^n$ . The resulting filter is of size  $(N + N - 1)^n$  and



**Fig. 1.** The computational load of filtering, i.e. the number of nonzero coefficients, can be reduced by designing and implementing the impulse response  $\tilde{f}(\xi)$  using cascaded subfilters  $h_k(\xi)$ ,  $k = 1, 2, \dots, K$

the computational gain for this particular filter is  $(N + N - 1)^n / (2N^n)$ . The main problem is how to decompose the desired filter response into subfilters that accurately enough can be described using sparsely scattered filter coefficients. For certain classes of filters, like for instance Cartesian separable filters, there is natural way to decompose the filters. But multidimensional filters are in general Cartesian nonseparable and no general factorization theorem exists for decomposition into sparse subfilters. Still heuristic approaches show examples of efficient decompositions of Cartesian nonseparable filters into cascaded subfilters (see e.g. [7]).

**3.1 Objective Functions**

Replacing  $\tilde{F}(\mathbf{u})$  in Eq. 2, 3 with the product of all subfilter frequency responses  $H_k(\mathbf{u})$  yields the least squares objectives Eq. 2 valid for arbitrary choice of subfilters. To simplify notation, the impulse response  $\tilde{f}(\xi)$  in Eq. 3 is now expressed as  $\tilde{f}(\xi) = \mathcal{F}^{-1}\{\tilde{F}(\mathbf{u})\}$ , i.e. the inverse Fourier transform of the frequency response.

$$\alpha(\mathbf{c}) = \int_{\mathbb{U}} W(\mathbf{u}) \left| F(\mathbf{u}) - \prod_k H_k(\mathbf{u}) \right|^2 d\mathbf{u} \tag{5}$$

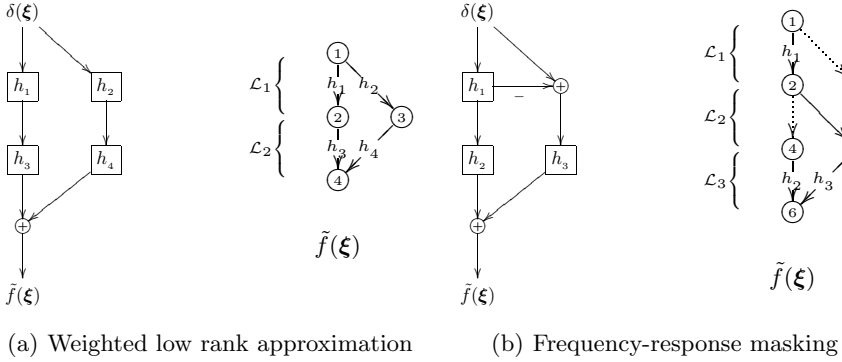
$$\beta(\mathbf{c}) = \sum_{\mathbb{Z}^n} w(\xi) \left| f(\xi) - \mathcal{F}^{-1}\left\{ \prod_k H_k(\mathbf{u}) \right\} \right|^2 \tag{6}$$

**4 Filter Networks**

There are three fundamental properties of the filter networks that contribute to computational efficiency. Firstly, intermediary results may contribute to multiple output, when designing a set of filters. Then, cascaded subfilters admit a lower number of filter coefficients compared to a direct implementation. Finally, sparse subfilters further decrease the number of nonzero filter coefficients, which contribute to a lower computational load. In software implementations a convolver that exploits sparsity is required [12].

**4.1 Graph Representation**

A general FIR filter network can be represented as a directed acyclic graph  $\mathcal{G} = (\mathcal{S}, \mathcal{H})$ , where the nodes  $s \in \mathcal{S}$  are summation points and an arc  $(s_i, s_j) \in \mathcal{H} \subseteq \mathcal{S} \times \mathcal{S}$  are a subfilter connecting  $s_i$  and  $s_j$ . Fig. 2 shows two small examples of such graph representations.



**Fig. 2.** To the left in both (a) and (b) the standard representation of filter networks is shown. The corresponding graphs are shown to the right. Subfilters  $h$  are represented by arcs and the nodes are summation points. The layers are denoted  $\mathcal{L}_k$ . The frequency-response masking example requires the use of dummy filters  $H(\mathbf{u}) = 1$  (the dotted arcs), to be able to draw a graph structured in layers. Weighted low rank approximations for 2-D are implemented as a sum of parallel branches with cascaded subfilters. The second order low rank approximation, i.e. two branches, is represented as a filter network with two layers

Elementary graph theory defines two nodes  $s_i, s_j$  as *adjacent*, when  $(s_i, s_j)$  is an arc. The entire graph can then be described by an adjacency matrix  $\mathbf{A}$ , with elements  $a_{ij}$  defined by Eq. 7.

$$a_{ij} = \begin{cases} 1, & (s_i, s_j) \in \mathcal{H} \\ 0, & (s_i, s_j) \notin \mathcal{H} \end{cases} \quad (7)$$

A *path*  $P$  is defined as a sequence of distinct nodes  $s_1, s_2, \dots, s_k$  such that  $s_i$  and  $s_{i+1}$  are adjacent for all  $i = 1, 2, \dots, k - 1$ . The length of a path  $P$  is for unweighted graphs defined as the number of arcs in  $P$ . Thus the adjacency matrix  $\mathbf{A}$  describes the paths of length 1 between every pair of nodes. Further on  $\mathbf{A}^k$  describes the all paths of length  $k$  between any pair of nodes. Consequently a path matrix  $\mathbf{P}$  as in Eq. 8 contains all paths between every pair of nodes.

$$\mathbf{P} = \sum_{k=0}^{\infty} \mathbf{A}^k \quad (8)$$

Let us now instead of just saying there is a relation between two nodes, label the relation  $(s_i, s_j)$  by  $H(\mathbf{u})$  representing the transfer function from node  $s_i$  to node  $s_j$ . Each element  $a_{ij}, p_{ij}$  in  $\mathbf{A}, \mathbf{P}$  then represent the transfer function between the nodes  $s_i$  and  $s_j$ . When studying filter networks, these transfer functions are of great interest, especially the ones relating the input node to the output nodes. For the small network example in Fig. 2(a),  $\mathbf{A}$  and  $\mathbf{P}$  is given by Eq. 9. For all examples in this paper row-wise numbering of the nodes from left to right is used as shown in Fig. 2.

$$\mathbf{A} = \begin{bmatrix} 0 & H_1 & H_2 & 0 \\ 0 & 0 & 0 & H_3 \\ 0 & 0 & 0 & H_4 \\ 0 & 0 & 0 & 0 \end{bmatrix} \quad \mathbf{P} = \begin{bmatrix} 1 & H_1 & H_2 & H_1H_3 + H_2H_4 \\ 0 & 1 & 0 & H_3 \\ 0 & 0 & 1 & H_4 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (9)$$

The transfer functions, with the numbering used, from the input to the output are represented by the rightmost elements on the first row  $p_{1o}$ , where  $o$  denotes the indices of the output nodes. For the example in Fig. 2(a) the output  $\tilde{F}(\mathbf{u})$  given by Eq. 10 is found as element  $p_{14}$  in  $\mathbf{P}$ .

$$\tilde{F}(\mathbf{u}) = \mathcal{F}\{\tilde{f}(\boldsymbol{\xi})\} = H_1(\mathbf{u})H_3(\mathbf{u}) + H_2(\mathbf{u})H_4(\mathbf{u}) = p_{14} \quad (10)$$

Clearly there is a more compact representation for the filter network output  $p_{1o}$ . For layered structured networks  $p_{1o}$  is obtained by decomposing  $\mathbf{A}$  into  $\mathbf{A}_k$ , where  $\mathbf{A}_k$  denotes the adjacency between nodes from layer  $k$  to layer  $k + 1$ . The reverse relations is not necessary, since there are no relations between nodes from layer  $k + 1$  to layer  $k$ . Thus  $\mathbf{A}_k$  is not quadratic and contains fewer zero elements. The output  $\tilde{\mathbf{F}} = [\tilde{F}_1(\mathbf{u}), \tilde{F}_2(\mathbf{u}), \dots, \tilde{F}_K(\mathbf{u})]^T$  of the design example is then given by Eq. 11.

$$\tilde{\mathbf{F}} = \mathbf{A}_2^T \mathbf{A}_1^T = [H_3 \ H_4] \begin{bmatrix} H_1 \\ H_2 \end{bmatrix} = H_1(\mathbf{u})H_3(\mathbf{u}) + H_2(\mathbf{u})H_4(\mathbf{u}) \quad (11)$$

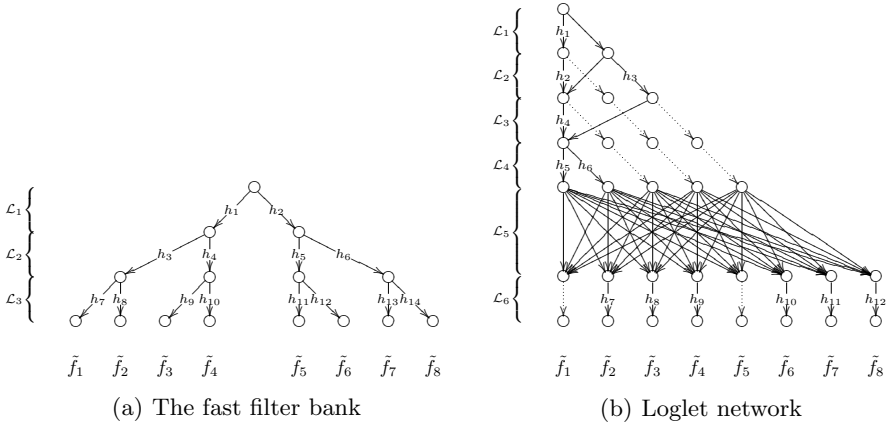
The compact representation can be used for arbitrary filter networks, since a layered structure can be obtained by inserting dummy filters with transfer function  $H(\mathbf{u}) = 1$  to extend all paths to have an equal length. The example in Fig. 2(b) requires dummy filters to be represented compactly as in Eq. 12.

$$\tilde{\mathbf{F}} = \mathbf{A}_3^T \mathbf{A}_2^T \mathbf{A}_1^T = [H_2 \ H_3] \begin{bmatrix} 1 & 0 \\ -1 & 1 \end{bmatrix} \begin{bmatrix} H_1 \\ 1 \end{bmatrix} = H_1(\mathbf{u})H_2(\mathbf{u}) + (1 - H_1(\mathbf{u}))H_3(\mathbf{u}) \quad (12)$$

## 4.2 Network Examples

The graph representation forms a general framework and many approaches fit within this framework. A few samples, conceptually close to the filter network approach in [1] are given. Firstly, frequency-response masking and the fast filter bank are briefly presented as two closely related 1-D samples. Then the widely used weighted low rank approximation constitute an example of decomposition of 2-D filters into sparse subfilters. Finally a 2-D version of the 3-D loglet network in [2], forms an example to show the use of graph representation on a larger filter network.

The frequency-response technique [10] can be thought of as generalizations of the IFIR approach. Frequency-response masking (FRM), shown in Fig. 2(b) allows an even sparser representation compared to the IFIR technique. This is achieved by reusing the filter response  $h_1(\boldsymbol{\xi})$  forming  $1 - h_1(\boldsymbol{\xi})$  (if causality is ignored). Applying interpolators  $h_2, h_3$  to each of these subfilter output filters  $F(\mathbf{u})$  in Eq. 12 with sharp transition bands with arbitrary bandwidth can be designed.



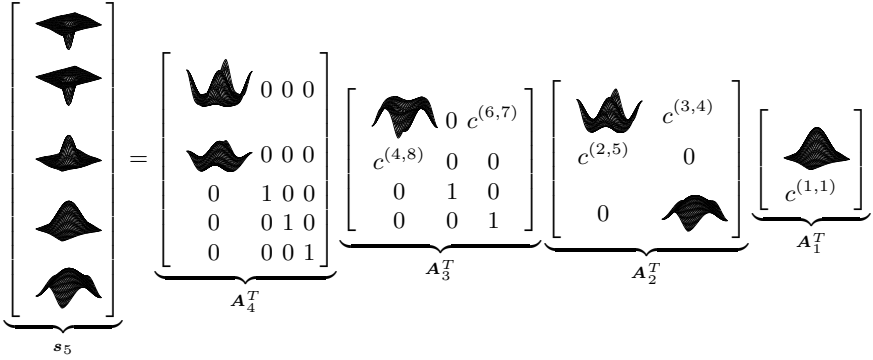
**Fig. 3.** The fast filter bank to the left is a single-rate system implemented as a tree-structured network. Here an 8 channel example is shown. This structure is also equivalent to a graph representation of an 8-point fast Fourier transform butterfly. The 2-D loglet network is represented by the network with 6 layers to the right. Arcs not annotated are single coefficient subfilters or dummy filters (dotted)

FRM can be applied in a multi-stage approach, by recursively designing the interpolators using FRM. It is also possible to analyze the branches separately i.e. having a multiple output network by not performing the summation after filtering with  $h_2, h_3$  in Fig. 2(b). This approach is used to derive the tree-structured fast filter bank (FFB) for efficient convolution in [13]. In fact, also the fast Fourier transform butterfly implemented with only 2 sparsely scattered subfilter coefficients (see e.g. [14]) can be represented using this structure shown in Fig. 3(a).

Most work on efficient convolution in multidimensional applications concerns factorization of the  $n$ -D desired frequency responses to achieve approximations in branching network structures using 1-D subfilters (see Fig. 2(a)). Weighted low rank approximations [15, 16] using singular value decomposition (SVD) to find the desired frequency responses for the 1-D subfilters is the most common technique. The parallel branches corresponding to the largest singular values then forms the implemented filter as in Eq. 11. Due to the lack of non-iterative SVD for  $n$  larger than 2 most research on WLRA is limited to 2-D.

The loglet network presented is represented by the graph in Fig. 3(b). The output  $\tilde{\mathbf{F}} = [\tilde{F}_1(\mathbf{u}), \tilde{F}_2(\mathbf{u}), \dots, \tilde{F}_8(\mathbf{u})]^T$  is given in Eq. 13 and constitute a basis for extracting features like orientation, velocity and local structure [17] in two different scales. The upper part of the network, from input to the output of layer  $\mathcal{L}_4$ , forms 5 radial basis functions denoted  $\mathbf{s}_5$  in Eq. 13, 14. The 5 basis functions and their relation to preceding subfilters are visualized in Fig. 4. The notation  $c^{(i,j)}$ , used represent single coefficient subfilters from node  $s_i$  to node  $s_j$ .

$$\tilde{\mathbf{F}} = \mathbf{A}_6^T \mathbf{A}_5^T \underbrace{\mathbf{A}_4^T \mathbf{A}_3^T \mathbf{A}_2^T \mathbf{A}_1^T}_{\mathbf{s}_5} \quad (13)$$



**Fig. 4.** The input to layer  $\mathcal{L}_5$  in the loglet network in Fig. 3(b) is computed as in Eq. 14. The transfer functions  $H_k(\mathbf{u})$  in these matrices are here visualized in the Fourier domain

The network output are then composed by filtering a linear combination (layer  $\mathcal{L}_5$ ) of these 5 basis functions with directional filters in layer  $\mathcal{L}_6$ . This computation is shown in Fig. 5.

$$s_5 = \begin{bmatrix} H_5 & 0 & 0 & 0 \\ H_6 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} H_4 & 0 & c^{(6,7)} \\ c^{(4,8)} & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} H_2 & c^{(3,4)} \\ c^{(2,5)} & 0 \\ 0 & H_3 \end{bmatrix} \begin{bmatrix} H_1 \\ c^{(1,1)} \end{bmatrix} \quad (14)$$

### 4.3 General Design Objectives

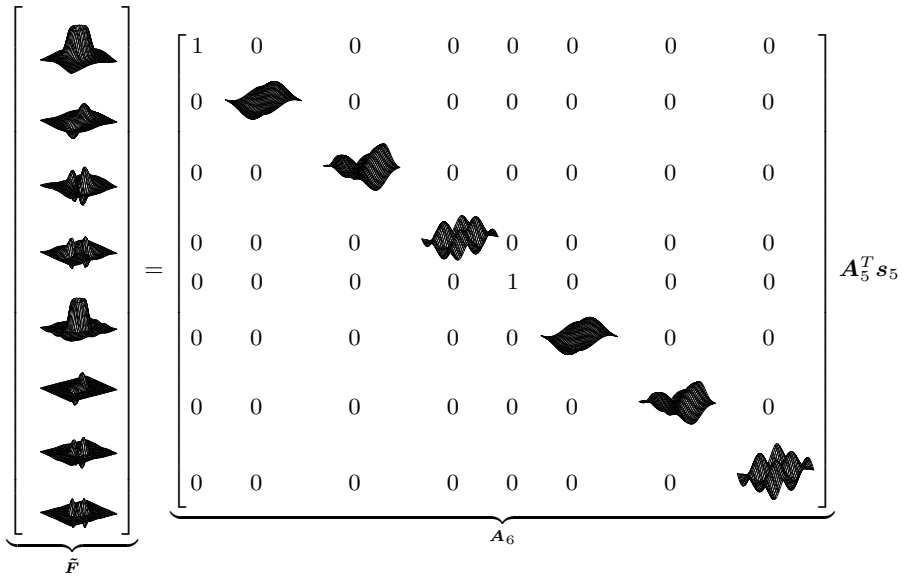
Actually, the problem of designing layered filter networks is similar to that of designing cascaded filters. Each layer in the filter network can be thought of as one subfilter in a filter sequence. As a consequence of the graph representation the objectives  $\alpha(\mathbf{c})$  in Eq. 5,  $\beta(\mathbf{c})$  in Eq. 6 used for cascaded subfilters can be generalized to represent filter networks in Eq. 15, 16. Note that the argument order, using the product operator can not be changed since  $\mathbf{A}_i$  and  $\mathbf{A}_j$  are not commutative.

$$\alpha(\mathbf{c}) = \int_{\mathbf{U}} \left( \mathbf{F} - \prod_k \mathbf{A}_k \right)^T \mathbf{W} \left( \mathbf{F} - \prod_k \mathbf{A}_k \right) d\mathbf{u} \quad (15)$$

$$\beta(\mathbf{c}) = \left( \mathbf{f} - \mathcal{F}^{-1} \left\{ \prod_k \mathbf{A}_k \right\} \right)^T \mathbf{w} \left( \mathbf{f} - \mathcal{F}^{-1} \left\{ \prod_k \mathbf{A}_k \right\} \right) \quad (16)$$

Sequential convolution fits within this approach, since cascaded filters constitute a directed acyclic graph, where each  $\mathbf{A}_k$  is a  $1 \times 1$  matrix. Note that Eq. 15, 16 is only valid for layered structured networks, but since all filter networks can be redrawn in such a way this is not a limitation.





**Fig. 5.** The output  $\tilde{F}$  of the loglet network in 3(b) is given by Eq. 13. Here the frequency responses are showing how the directional filters in layer  $\mathcal{L}_6$  forms the output from linear combinations of the input to layer  $\mathcal{L}_5$  shown in Fig. 4

## 5 Discussion

Design of filter sets using filter networks offers a manifold of opportunities to increase the computational efficiency. If similarities between the filters in the set can be exploited, subfilters can contribute to multiple output. To fully exploit this property it is necessary to search for a good network structure.

Choosing the network structure optimally is however a very difficult task, since it require joint optimization of the network structure, the discrete spatio-temporal positions of the filter coefficients and the coefficient values.

The graph representation presented in this paper forms a framework for design of filter networks, which incorporates the network structure in the design objectives and removes the requirement of the layered network structure previously used. The framework simplifies analysis and visualization of how the network structure influences the objective functions.

## References

1. M. Andersson, J. Wiklund, and H. Knutsson. Filter networks. In *Proceedings of Signal and Image Processing (SIP'99)*, Nassau, Bahamas, October 1999. IASTED. Also as Technical Report LiTH-ISY-R-2245.
2. B. Svensson, M. Andersson, and H. Knutsson. Filter networks for efficient estimation of local 3d structure. In *Proceedings of the IEEE-ICIP*, Genoa, Italy, September 2005.

3. J. H. McClellan and T. W. Parks. A unified approach to the design of optimum FIR linear phase digital filters. *IEEE Trans. Circuit Theory*, CT-20:697–701, 1973.
4. D. W. Tufts and J. T. Francis. Designing digital low pass filters - Comparison of some methods and criteria. *IEEE Trans. Audio Electroacoust.*, AU-18:487–494, August 1970.
5. P. Vaidyanathan and T. Nguyen. Eigenfilters: A new approach to least-squares FIR filter design and applications including nyquist filters. *IEEE Transactions on Circuits and Systems*, 34(1):11–23, 1987.
6. J.L. Aravena and Guoxiang Gu. Weighted least mean square design of 2-d fir digital filters: the general case. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 44(10):2568 – 2578, Oct. 1996.
7. H. Knutsson, M. Andersson, and J. Wiklund. Advanced filter design. In *Proceedings of the Scandinavian Conference on Image analysis*, Greenland, June 1999. SCIA.
8. T. Saramäki and A. Fam. Subfilter approach for designing efficient fir filters. In *Proceedings IEEE International Symposium on Circuits and Systems*, pages 2903–2915, June 1988.
9. J. Kaiser and R. Hamming. Sharpening the response of a symmetric nonrecursive filter by multiple use of the same filter. *IEEE Transaction on Acoustics, Speech, and Signal Processing*, ASSP-25:415–422, Oct. 1977.
10. Y. C. Lim. Frequency-response masking approach for the synthesis of sharp linear phase digital filters. *IEEE Trans. Circuits and Systems*, CAS-33:357–364, Apr. 1986.
11. Y. Neuvo, Dong Cheng-Yu, and S. Mitra. Interpolated finite impulse response filters. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 32(3):563 – 570, June 1984.
12. J. Wiklund and H. Knutsson. A generalized convolver. Report LiTH-ISY-R-1830, Computer Vision Laboratory, SE-581 83 Linköping, Sweden, April 1996.
13. Yong Ching Lim and B. Farhang-Boroujeny. Analysis and optimum design of the ffb. In *Proceedings of IEEE International Symposium on Circuits and Systems (ISCAS'94)*, volume 2, pages 509 – 512, May 1994.
14. Y.C. Lim and B. Farhang-Boroujeny. Fast filter bank (ffb). *IEEE Transactions on Circuits and Systems II: Analog and Digital Signal Processing*, 39(5):316 – 318, May 1992.
15. J. Shanks, S. Treitel, and J. Justice. Stability and synthesis of two-dimensional recursive filters. *IEEE Transactions on Audio and Electroacoustics*, 20(2):115 – 128, June 1972.
16. R. Twogood and S. Mitra. Computer-aided design of separable two-dimensional digital filters. *IEEE Trans. Acoust. Speech, Signal Processing*, 25:165–169, 1977.
17. G. H. Granlund and H. Knutsson. *Signal Processing for Computer Vision*. Kluwer Academic Publishers, 1995. ISBN 0-7923-9530-1.