

Nonlinear Filtering

- Linear filters
 - Tend to blur edges and other image detail.
 - Perform poorly with non-Gaussian noise.
 - Result from Gaussian image and noise assumptions.
 - Images are not Gaussian.
- Nonlinear filter
 - Can preserve edges
 - Very effective at removing impulsive noise
 - Result from non-Gaussian image and noise assumptions.
 - Can be difficult to design.

Linear Filters

- Definition: A system $y = T[x]$ is said to be linear if for all $\alpha, \beta \in \mathbb{R}$

$$\alpha y_1 + \beta y_2 = T[\alpha x_1 + \beta x_2]$$

where $y_1 = T[x_1]$ and $y_2 = T[x_2]$.

- Any filter of the form

$$y_s = \sum_r h_{s,r} x_r$$

Homogeneous Filters

- Definition: A filter $y = T[x]$ is said to be homogeneous if for all $\alpha \in \mathbb{R}$

$$\alpha y = T[\alpha x]$$

- This is much weaker than linearity.
- Homogeneity is a natural condition for scale invariant systems.

Median Filter

- Let W be a window with an odd number of points.

- Then the median filter is given by

$$y_s = \text{median} \{x_{s+r} : r \in W\}$$

- Is the median filter:

- Linear?
- Homogeneous?

- Consider the 1-D median filter with a 3-point window.

x(m)	0	0	1	1,000	1	1	2	2
y(m)	?	0	1	1	1	1	2	?

Median Filter: Optimization Viewpoint

- Consider the median filter

$$y_s = \text{median} \{x_{s+r} : r \in W\}$$

and consider the following functional.

$$F(\theta) \triangleq \sum_{r \in W} |\theta - x_{s+r}|$$

- Then y_s solves the following optimization equation.

$$y_s = \arg \min_{\theta} F(\theta)$$

- Differentiating, we have

$$\begin{aligned} \frac{d}{d\theta} F(\theta) &= \frac{d}{d\theta} \sum_{r \in W} |\theta - x_{s+r}| \\ &= \sum_{r \in W} \text{sign}(\theta - x_{s+r}) \\ &\triangleq f(\theta) \end{aligned}$$

This expression only holds for $\theta \neq x_{s+r}$ for all $r \in W$.

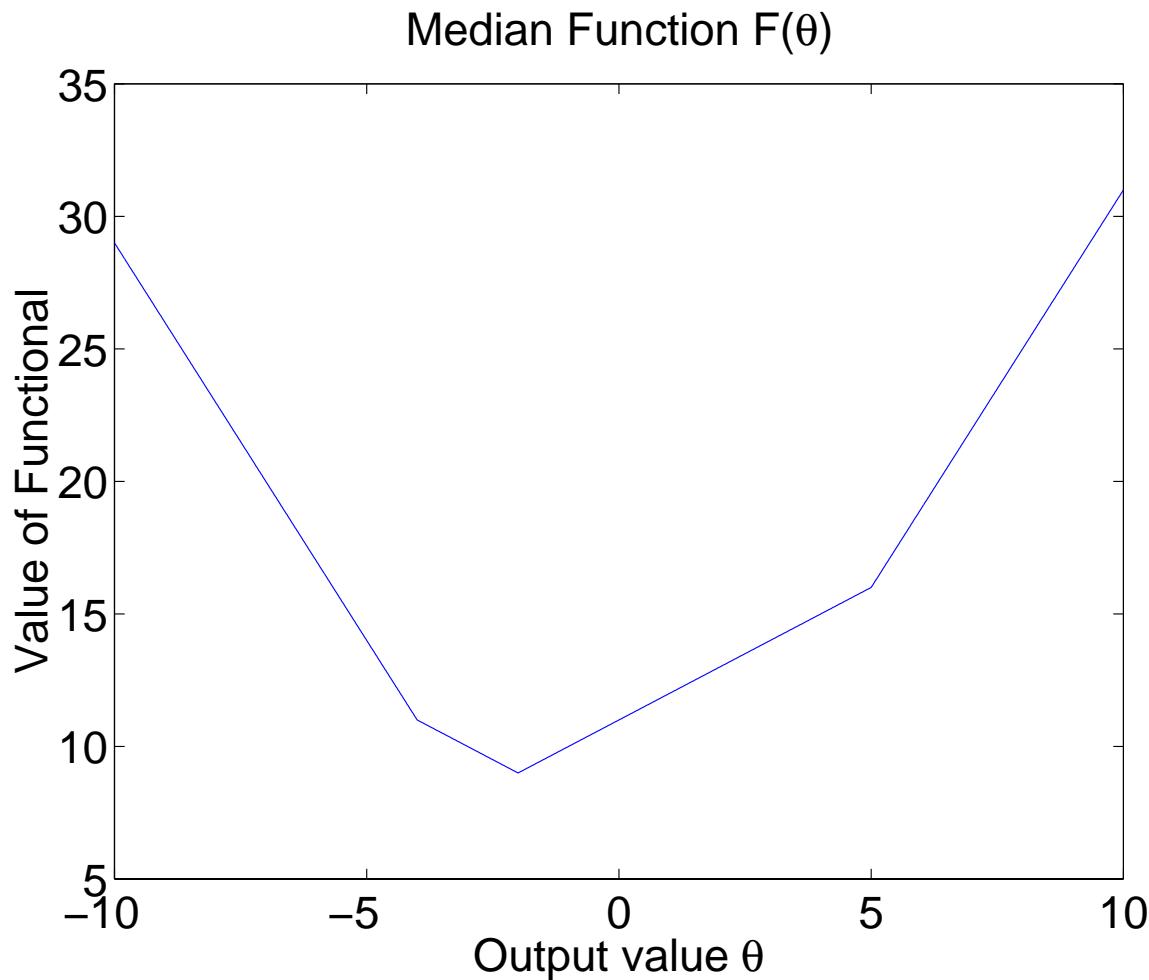
- So the solution falls at $\theta = x_{s*}$ such that

$$0 = \sum_{\substack{r \in W \\ r \neq (s*-s)}} \text{sign}(\theta - x_{s+r})$$

Example: Median Filter Function

- Consider a 1-D median filter
 - Three point window of $W = \{-1, 0, 1\}$
 - Inputs $[x(n-1), x(n), x(n+1)] = [-2, -4, 5]$.

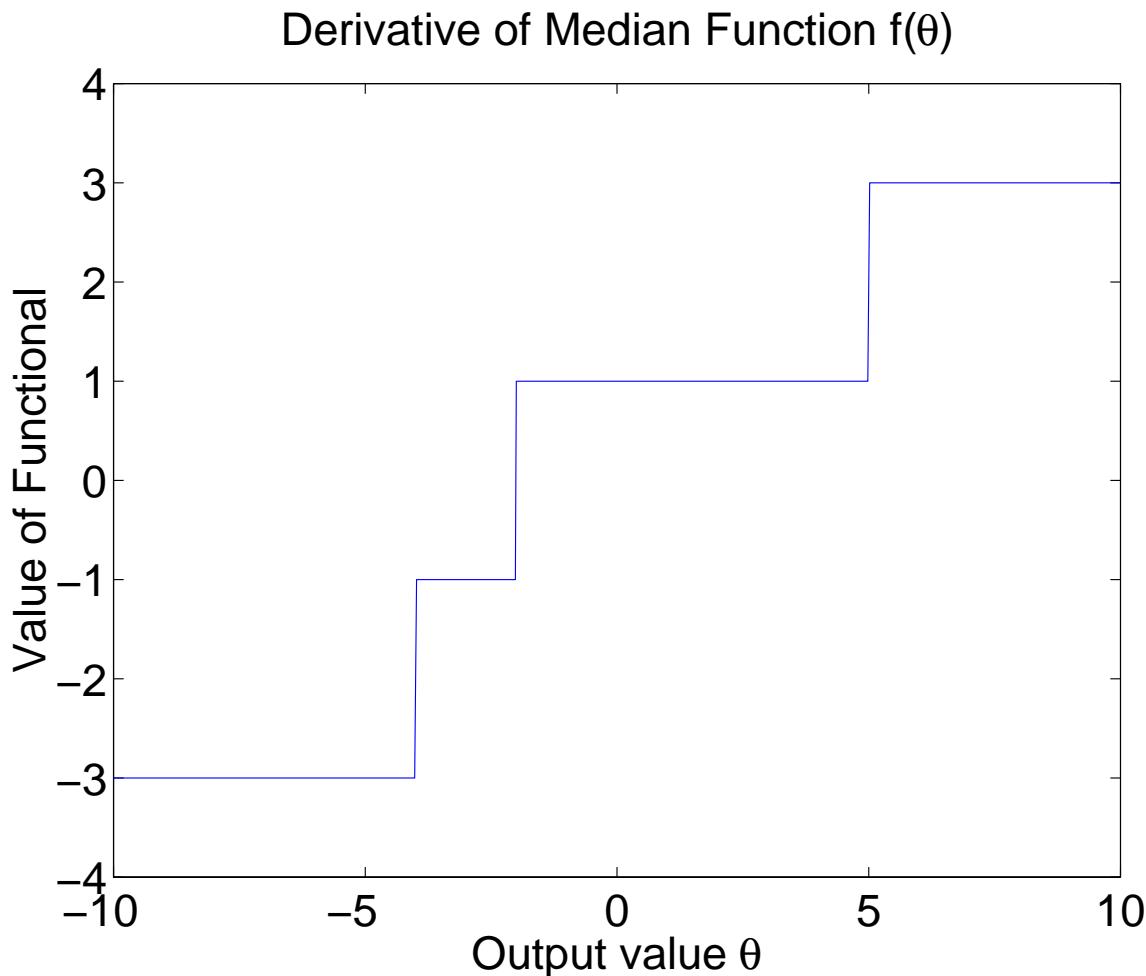
$$F(\theta) = \sum_{k=-1}^1 |\theta - x_{n+k}|$$



Example: Derivative of Median Filter Function

- Consider a 1-D median filter
 - Three point window of $W = \{-1, 0, 1\}$
 - Inputs $[x(n-1), x(n), x(n+1)] = [-2, -4, 5]$.

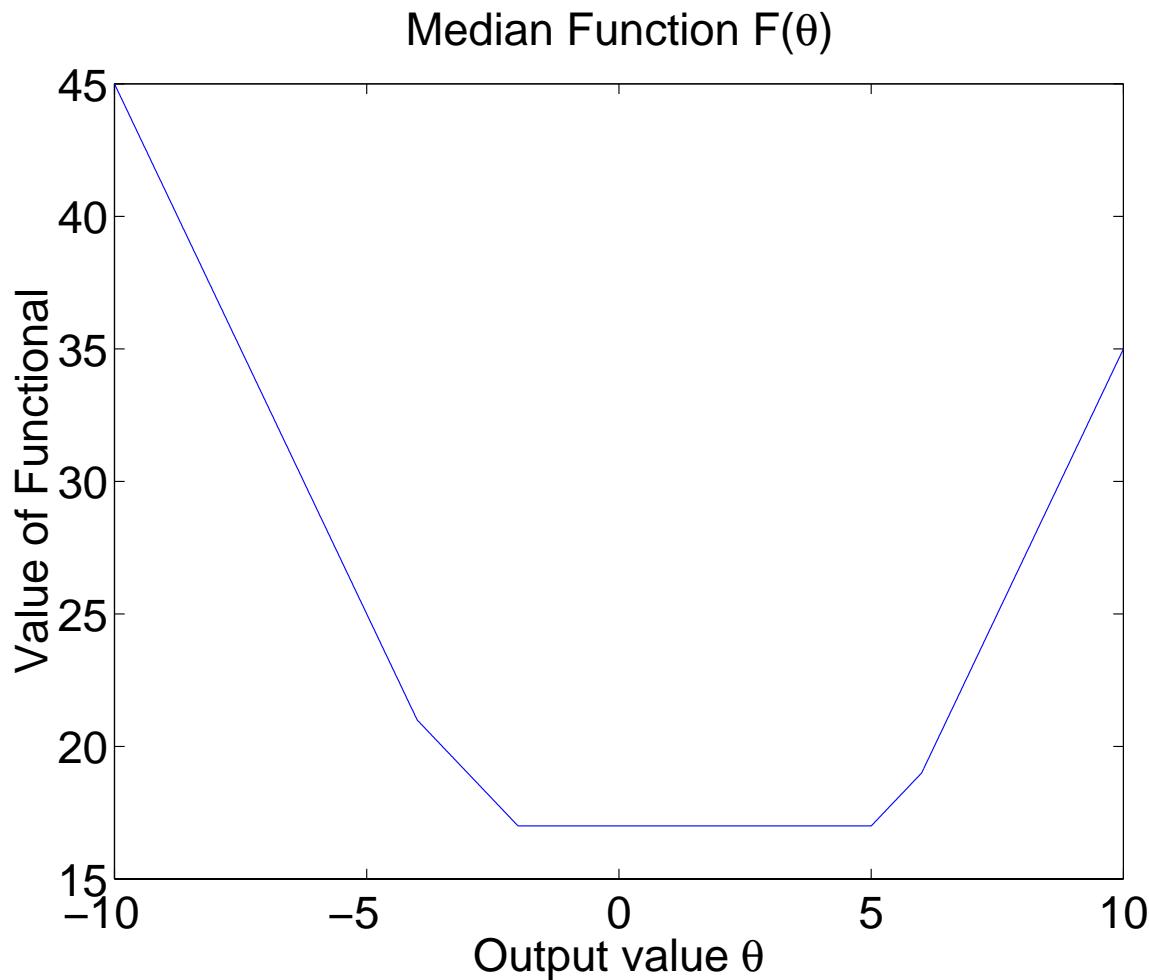
$$f(\theta) = \sum_{k=-1}^1 \text{sign}(\theta - x_{n+k})$$



Problem with an Even Number of Points

- Consider a 1-D median filter
 - Four point window of $W = \{-1, 0, 1, 2\}$
 - Inputs $[x(n-1), x(n), x(n+1), x(n+2)] = [-2, -4, 5, 6]$.
- Solution is not unique.

$$F(\theta) = \sum_{k=-1}^2 |\theta - x_{n+k}|$$



Weighted Median Filter

- Defined the functional

$$F(\theta) \triangleq \sum_{r \in W} a_r |\theta - x_{s+r}|$$

where a_r are weights assigned to each point in the window W .

- Weighted median is computed by

$$y_s = \arg \min_{\theta} \sum_{r \in W} a_r |\theta - x_{s+r}|$$

- Differentiating, we have

$$\begin{aligned} \frac{d}{d\theta} F(\theta) &= \frac{d}{d\theta} \sum_{r \in W} a_r |\theta - x_{s+r}| \\ &= \sum_{r \in W} a_r \text{sign}(\theta - x_{s+r}) \\ &\triangleq f(\theta) \end{aligned}$$

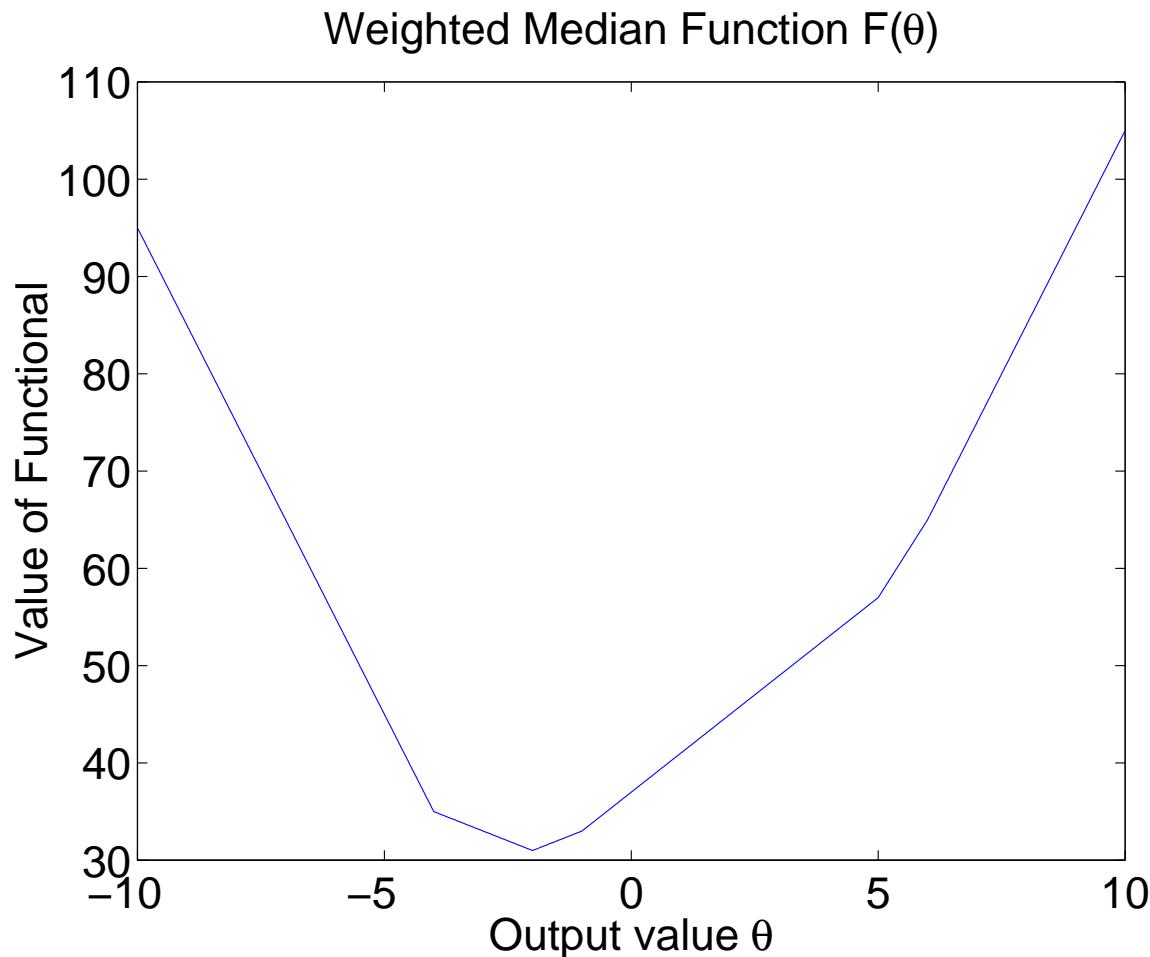
This expression only holds for $\theta \neq x_r$ for all $r \in W$.

- Need to find $s*$ such that $f(\theta)$ is “nearly” zero.

Example: Weighted Median Filter Function

- Consider a 1-D median filter
 - Five point window of $W = \{-2, -1, 0, 1, 2\}$
 - Inputs $[x(n-2), \dots, x(n+2)] = [6, -2, -4, 5, -1]$.
 - Weights $[a(-2), a(-1), a(0), a(1), a(2)] = [1, 2, 4, 2, 1]$.

$$F(\theta) = \sum_{k=-1}^1 a(k) |\theta - x_{n+k}|$$

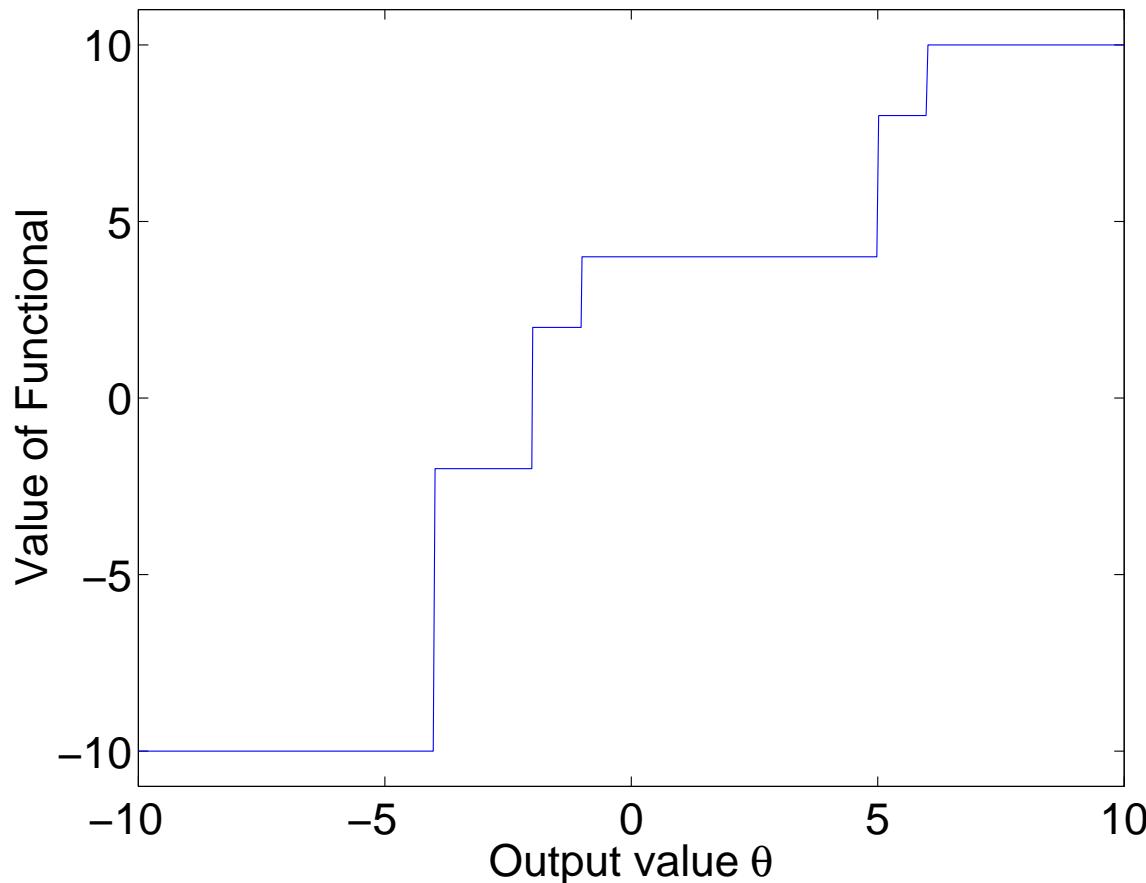


Example: Derivative of Median Filter Function

- Consider a 1-D median filter
 - Five point window of $W = \{-2, -1, 0, 1, 2\}$
 - Inputs $[x(n-2), \dots, x(n+2)] = [6, -2, -4, 5, -1]$.
 - Weights $[a(-2), a(-1), a(0), a(1), a(2)] = [1, 2, 4, 2, 1]$.

$$f(\theta) = \sum_{k=-1}^1 a(k) \operatorname{sign}(\theta - x_{n+k})$$

Derivative of Weighted Median Function $f(\theta)$



Computation of Weighted Median

1. Sort points in window.

- Let $x_{(1)} < x_{(2)} < \dots < x_{(p)}$ be the sorted values.
- These values are known as order statistics.
- Let $a_{(1)} < a_{(2)} < \dots < a_{(p)}$ be the **corresponding** weights.

2. Find i^* such that

$$\sum_{i=1}^{i^*} a_{(i)} > \sum_{i=i^*-1}^p a_{(i)}$$

and

$$\sum_{i=1}^{i^*-1} a_{(i)} < \sum_{i=i^*}^p a_{(i)}$$

3. Then the value $a_{(i^*)}$ is the weighted median value.

Comments on Weighted Median Filter

- Weights may be adjusted to yeild the “best” filter.
- Largest and smallest values are ignored.
- Same as median filter for $a_r = 1$.