Piecewise Polynomial Convolutions and Orthogonal Projections

Fabian Andres Prada Niño

IMPA

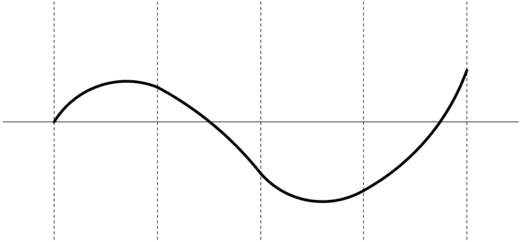
Part I

Piecewise Polinomial Function

- **Definition** $f : \mathbb{R} \to \mathbb{R}$ is said *piecewise polynomial function* if there exists reals $r_1 < r_2 < \ldots < r_{n+1}$ and polynomials $p_1(x), p_2(x) \ldots p_n(x)$, such that $supp(f) = [r_1, r_{n+1}]$, and for $k = 1 \ldots n$ we have $f \equiv p_k$ in the interval $[r_k, r_{k+1}]$.

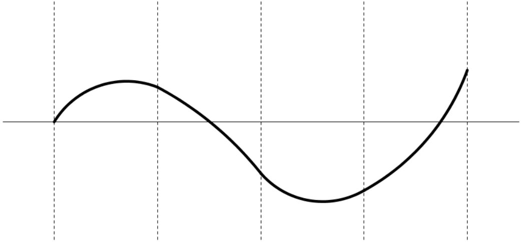
Piecewise Polinomial Function

- **Definition** $f : \mathbb{R} \to \mathbb{R}$ is said *piecewise polynomial function* if there exists reals $r_1 < r_2 < \ldots < r_{n+1}$ and polynomials $p_1(x), p_2(x) \ldots p_n(x)$, such that $supp(f) = [r_1, r_{n+1}]$, and for $k = 1 \ldots n$ we have $f \equiv p_k$ in the interval $[r_k, r_{k+1}]$.



Piecewise Polinomial Function

- **Definition** $f : \mathbb{R} \to \mathbb{R}$ is said *piecewise polynomial function* if there exists reals $r_1 < r_2 < \ldots < r_{n+1}$ and polynomials $p_1(x), p_2(x) \ldots p_n(x)$, such that $supp(f) = [r_1, r_{n+1}]$, and for $k = 1 \ldots n$ we have $f \equiv p_k$ in the interval $[r_k, r_{k+1}]$.



- **Definition** The pair $((r_1, \ldots, r_{n+1}); (p_1, \ldots, p_n))$ will be called the canonical representation of f.

Problem Statement

Problem Statement Given f and g piecewise polynomial in canonical representation, find the canonical representation of $f * g$.

Polynomials Supports

Question: How many polynomial pieces does $f * g$ have?

Polynomials Supports

Question: How many polynomial pieces does $f * g$ have?
It depends on the spacing of polynomial supports...

Polynomials Supports

Question: How many polynomial pieces does $f * g$ have?
It depends on the spacing of polynomial supports...

- If polynomials supports in f and g are uniformly and equally spaced , $f * g$ will have $n + m$ distinct polynomial pieces.

Polynomials Supports

Question: How many polynomial pieces does $f * g$ have?
It depends on the spacing of polynomial supports...

- If polynomials supports in f and g are uniformly and equally spaced , $f * g$ will have $n + m$ distinct polynomial pieces.
- If polynomials supports of f and g are in general spacing, $f * g$ will have $n * m$ distinct polynomial pieces.

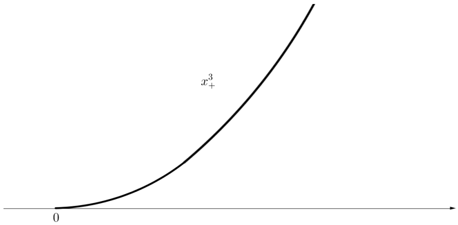
Convolution Strategy

Main Concepts

Convolution Strategy

Main Concepts

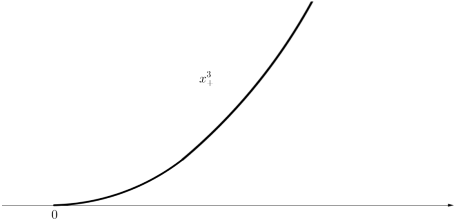
- **Power Step Functions**



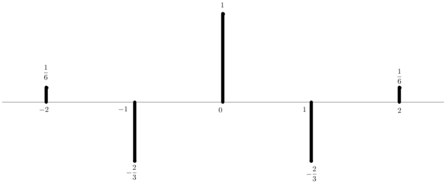
Convolution Strategy

Main Concepts

- **Power Step Functions**



- **Sequence of Impulses**



Power Step Functions

Definition: The function,

$$(x_+)^n = \begin{cases} x^n & \text{if } x > 0 \\ 0 & \text{if } x \leq 0 \end{cases}$$

will be called the power step function of order n.

A nice property

Proposition Let n and m be nonnegative integers, then

$$(x_+)^n * (x_+)^m = \frac{1}{\binom{n+m}{n}} (x_+)^{n+m+1}$$

Sequence of Impulses

Definition: Given $r = (r_1, \ldots, r_n)$ with $r_1 < r_2 < \ldots < r_n$, and $w = (w_1, \ldots, w_n)$ with w_k arbitrary, the function,

$$T(r, w)(x) = \sum_{k=1}^{n} w_k \delta(x - r_k),$$

will be called the *sequence of impulses* associated to (r, w). The vector r will be called the *knots* of the sequence, and the vector w will be called the *weights* of the sequence.

Example

The Bspline of order n can be easily represented as a convolution between a sequence of impulses and a power step function. Let:

$$r = \left(\left(0 - \frac{n+1}{2}\right), \left(1 - \frac{n+1}{2}\right), \ldots, \left(n+1 - \frac{n+1}{2}\right) \right)$$

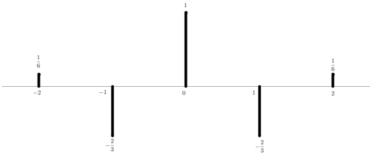
$$w = \frac{1}{n!} \left((-1)^0 \binom{n+1}{0}, (-1)^1 \binom{n+1}{1}, \ldots, (-1)^{n+1} \binom{n+1}{n+1} \right)$$

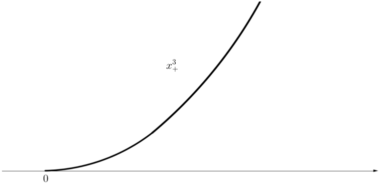
Then,

$$\beta_n = T(r, w) * (x_+)^n$$

Example

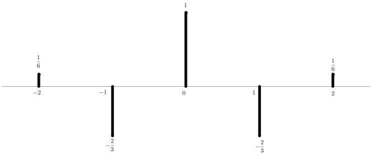
Convolved Functions:

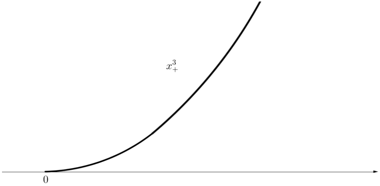




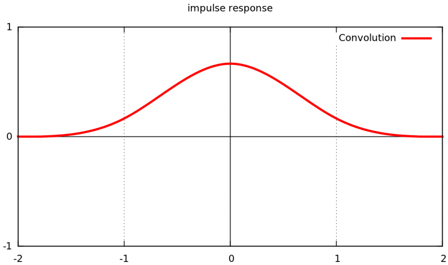
Example

Convolved Functions:





Convolution Result:



Overview of the convolution algorithm

- **Input:** f, g in canonical representation.

Overview of the convolution algorithm

- **Input:** f, g in canonical representation.
- 1) Convert f and g to power step representation.

Overview of the convolution algorithm

- **Input:** f, g in canonical representation.
- 1) Convert f and g to power step representation.
- 2) Convolve f and g in power step representation.

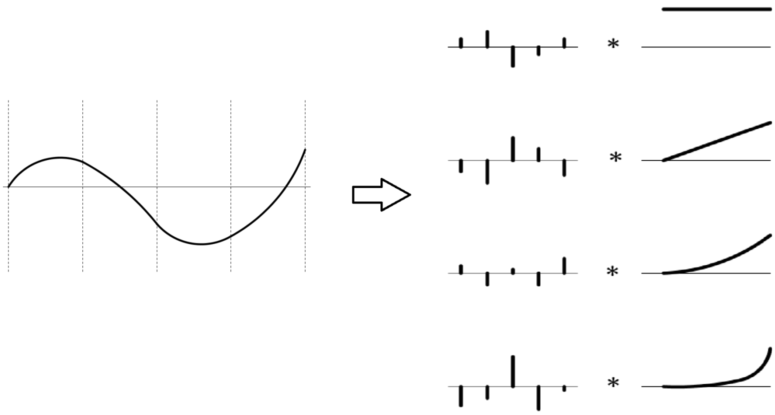
Overview of the convolution algorithm

- **Input:** f, g in canonical representation.
- 1) Convert f and g to power step representation.
- 2) Convolve f and g in power step representation.
- 3) Convert $f * g$ to canonical representation.

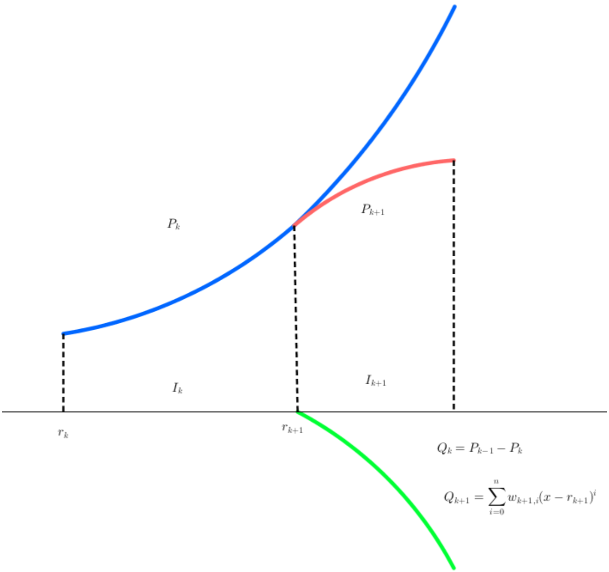
Overview of the convolution algorithm

- **Input:** f, g in canonical representation.
- 1) Convert f and g to power step representation.
- 2) Convolve f and g in power step representation.
- 3) Convert $f * g$ to canonical representation.
- **Output:** $f * g$ in canonical representation.

From Canonical to Power Step



From Canonical to Power Step



From Canonical to Power Step

Let

$$Q_{k+1}(x) = P_{k+1}(x) - P_k(x) = a_n x^n + \ldots + a_1 x + a_0,$$

Then,

$$\begin{aligned} Q_{k+1}(x) &= Q_{k+1}((x - r_{k+1}) + r_{k+1}) \\ &= \sum_{i=0}^{n} a_i ((x - r_{k+1}) + r_{k+1})^i \\ &= \sum_{i=0}^{n} \Big(\sum_{i \le m \le n} a_m \binom{m}{i} r_{k+1}^{m-i} \Big) (x - r_{k+1})^i \end{aligned}$$

Therefore,

$$w_{k+1,i} = \Big(\sum_{i \le m \le n} a_m \binom{m}{i} (r_{k+1})^{m-i} \Big)$$

Convolution

Proposition Given f piecewise polynomial of degree n there exists vectors $r, w_0, w_1, \ldots, w_n$ such that

$$f = \sum_{i=0}^{n} T(r, w_i) * (x_+)^i$$

Convolution

Proposition Given f piecewise polynomial of degree n there exists vectors $r, w_0, w_1, \ldots, w_n$ such that

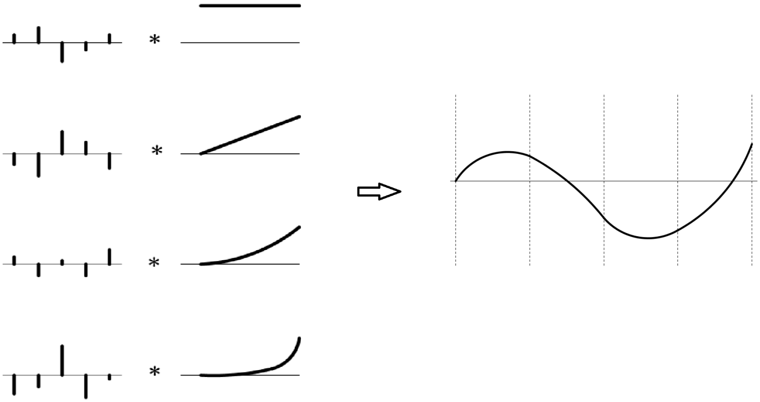
$$f = \sum_{i=0}^{n} T(r, w_i) * (x_+)^i$$

Corollary $f = \sum_{i=0}^{n} T(r, w_i) * (x_+)^i$, $g = \sum_{j=0}^{n} T(s, u_j) * (x_+)^j$

$$\Rightarrow f * g = \sum_{k=0}^{n+m+1} T_k * (x_+)^k,$$

Where $T_k = \sum_{i+j=k} \frac{1}{\binom{k}{i}} T(r, w_i) * T(s, u_j)$

From Power Step to Canonical



From Power Step to Canonical

Suppose $P_k(x) = a_n x^x + \ldots + a_i x^i + \ldots + a_0$ has already been calculated, then

$$P_{k+1}(x) = P_k(x) + \sum_{i=0}^{n} w_{k+1,i}(x - r_{k+1})^i$$

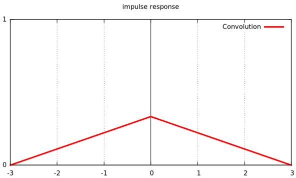
$$= \sum_{i=0}^{n} \Big(a_i + \sum_{i \le m \le n} w_{k+1,m} \binom{m}{i} (-r_{k+1})^{m-i} \Big) x^i,$$

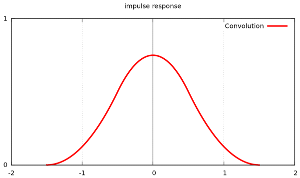
This last expression gives the values of the coefficients of P_{k+1}.

Some Results

Example 1

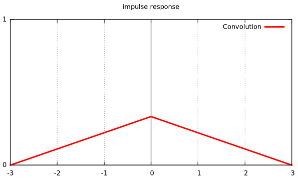
Convolved Functions:

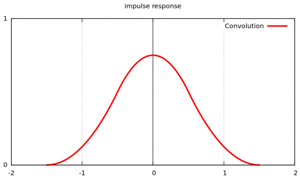




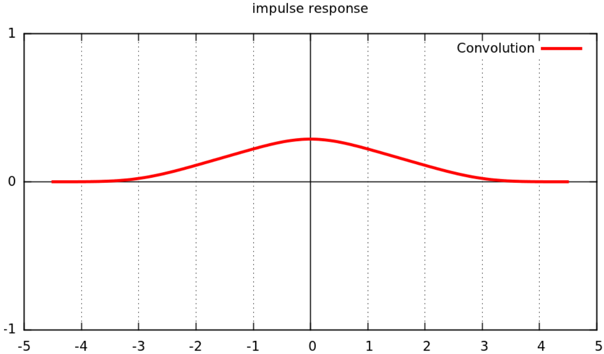
Example 1

Convolved Functions:



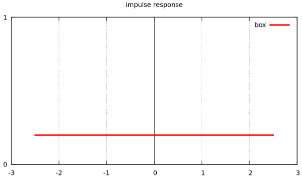


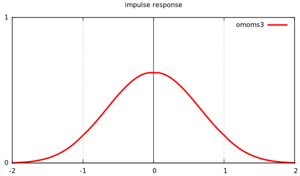
Convolution Result:



Example 2

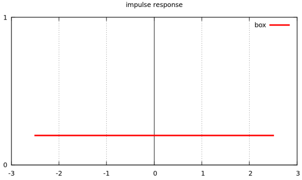
Convolved Functions:

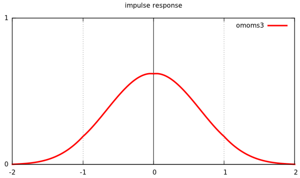




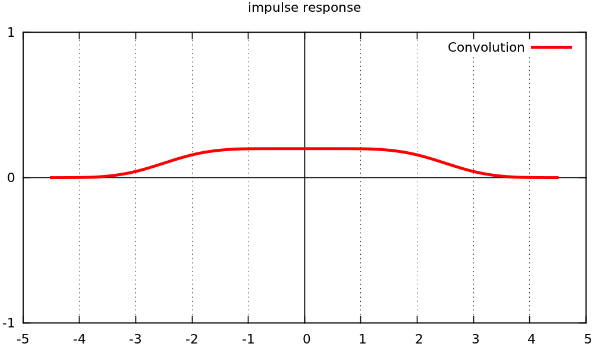
Example 2

Convolved Functions:





Convolution Result:



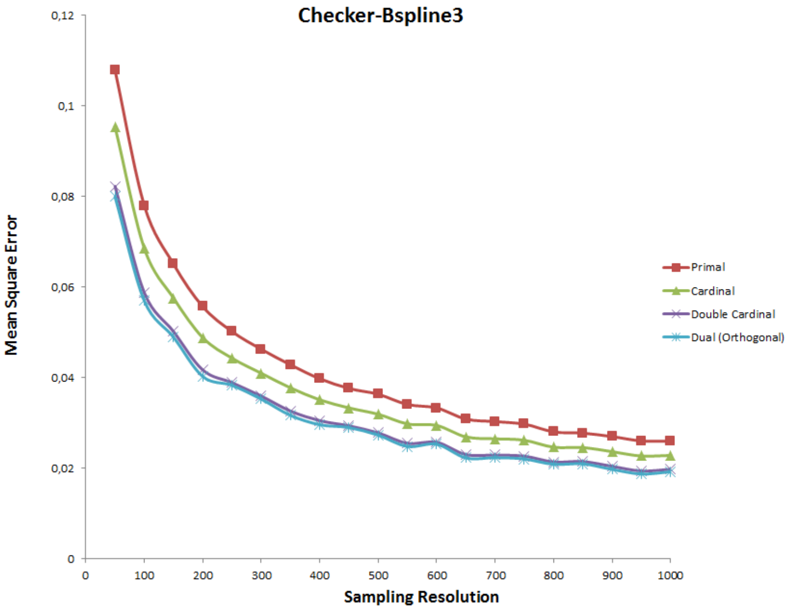
Part II

Experiment 1

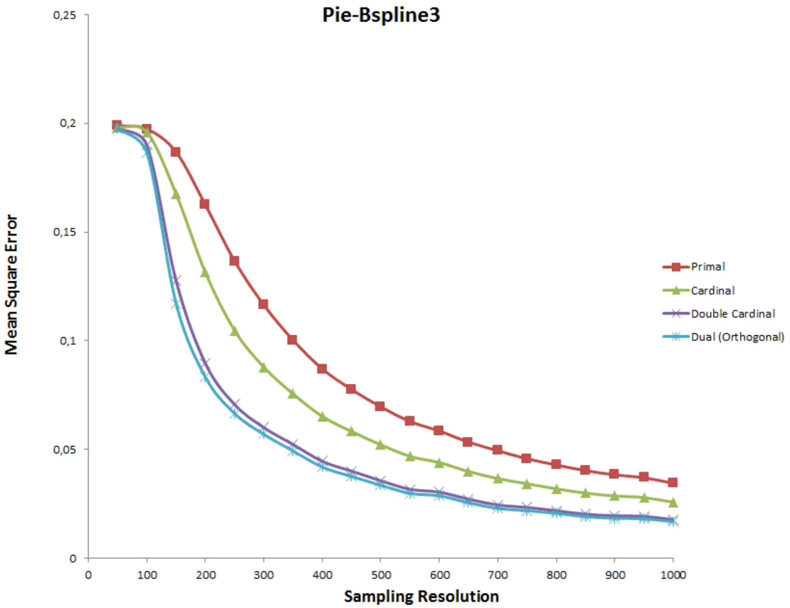
Input. Texture Function : f
Output. Reconstructed Signal : $\tilde{f} = [f * \varphi] * d * \varphi$.
Objective. Identify the quality of approximation that $\tilde{f}$ provides to f for a fixed type filter and some reconstruction methods.
Reconstruction Methods

- Primal: $d = \delta$.
- Cardinal: $d = [\varphi]^{-1}$.
- Double Cardinal: $d = [\varphi]^{-1} * [\varphi]^{-1}$.
- Dual (Orthogonal Projection): $d = [\varphi * \varphi^{\vee}]^{-1}$

Experiment 1. Results 1



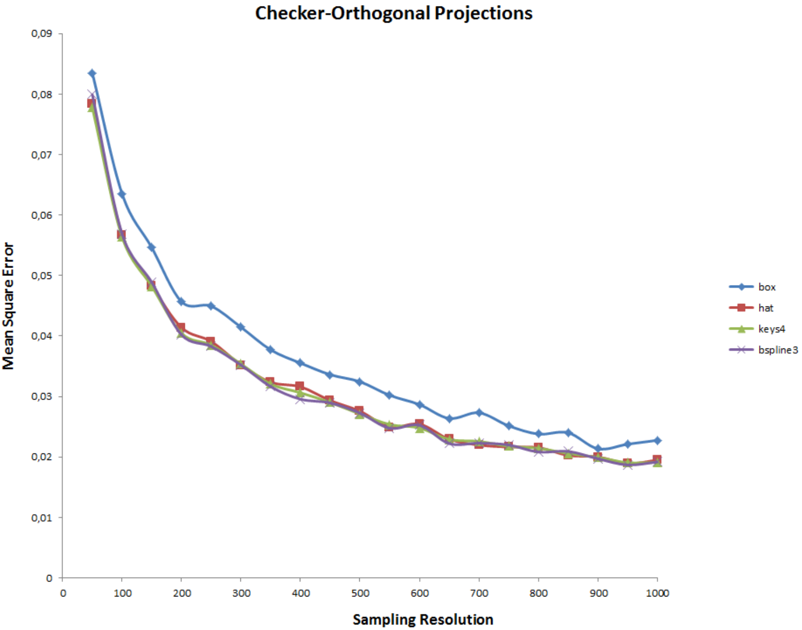
Experiment 1. Results 2



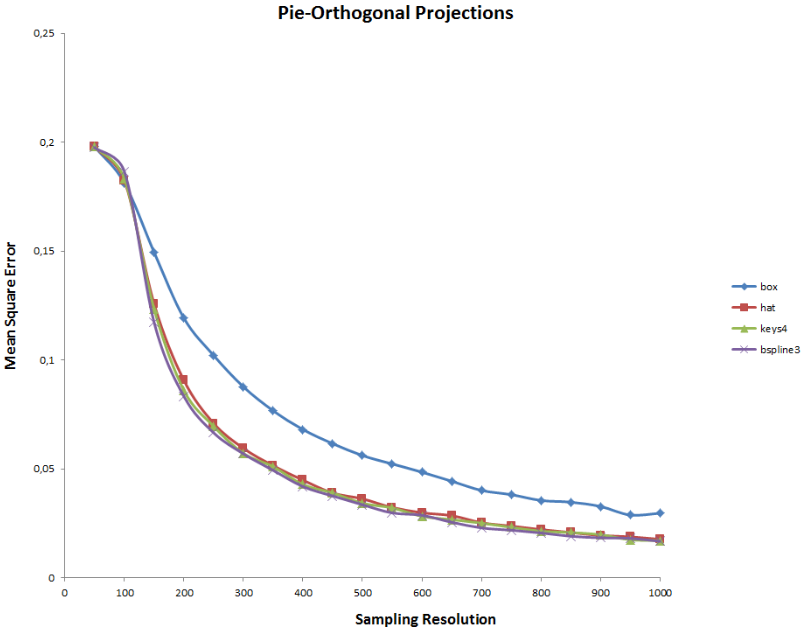
Experiment 2

Input. Texture Function : f
Output. Orthogonal Projection: $\tilde{f} = [f * \varphi] * [\varphi * \varphi^{\vee}]^{-1} * \varphi$.
Objective. Compare the quality of approximation that $\tilde{f}$ provides to f for some filters as a function of the sampling resolution.

Experiment 2. Results 1



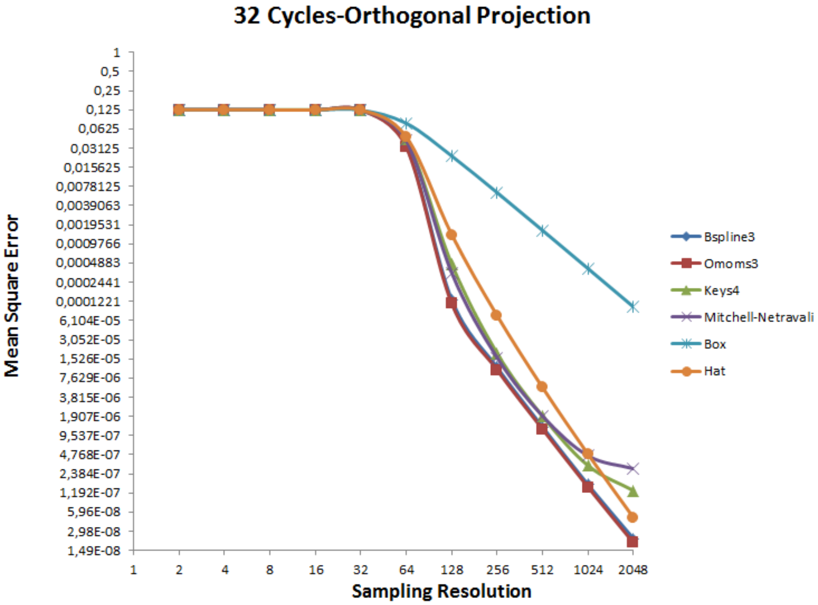
Experiment 2. Results 2



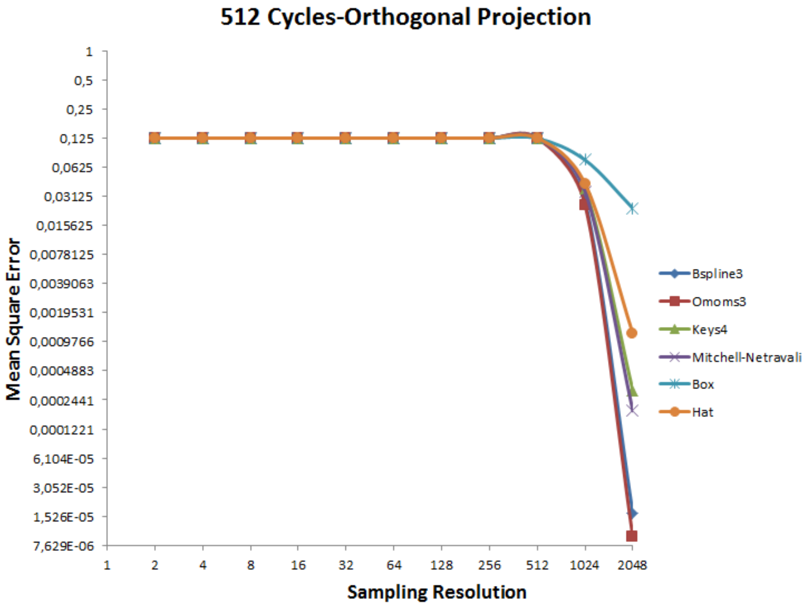
Experiment 3

Input. Fundamental Frequency: f
Output. Orthogonal Projection: $\tilde{f} = [f * \varphi] * [\varphi * \varphi^{\vee}]^{-1} * \varphi$.
Objective. Identify the order of approximation that $\tilde{f}$ provides to f for some filters as a function of the sampling resolution.

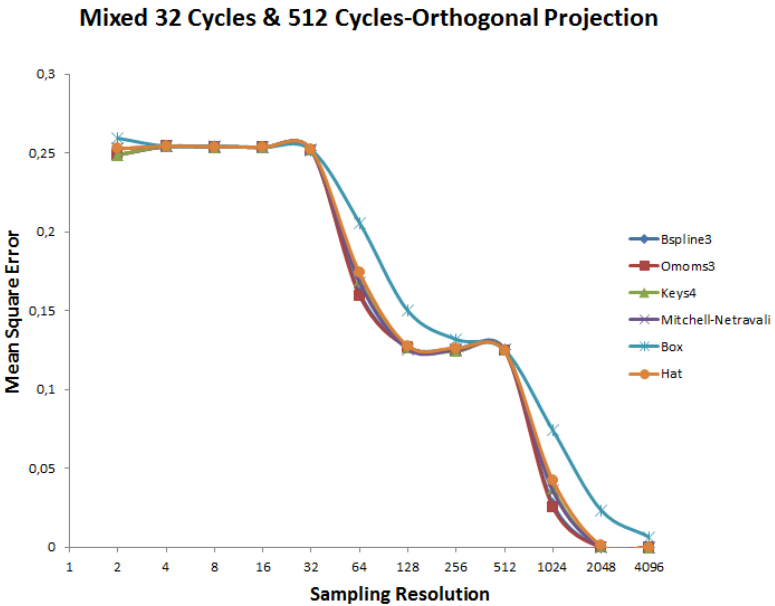
Experiment 3. Results 1



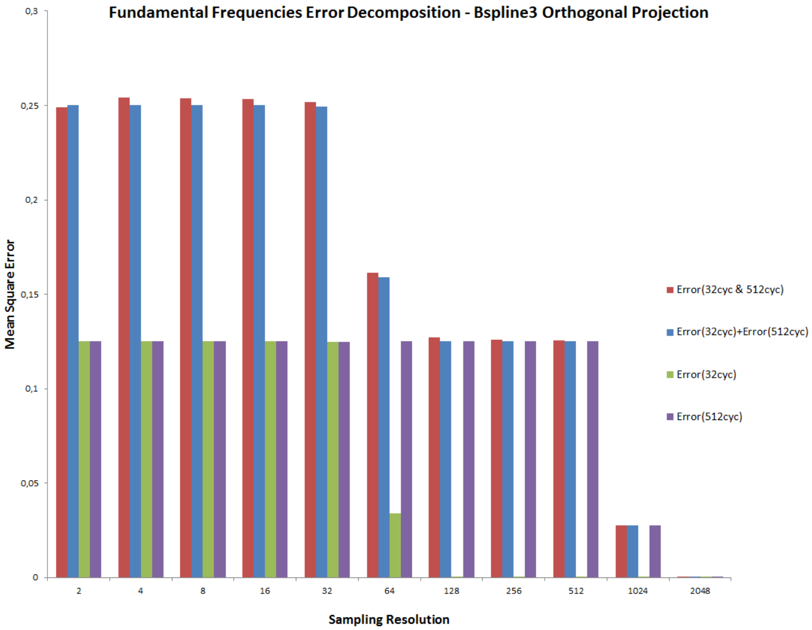
Experiment 3. Results 2



Experiment 3. Results 3



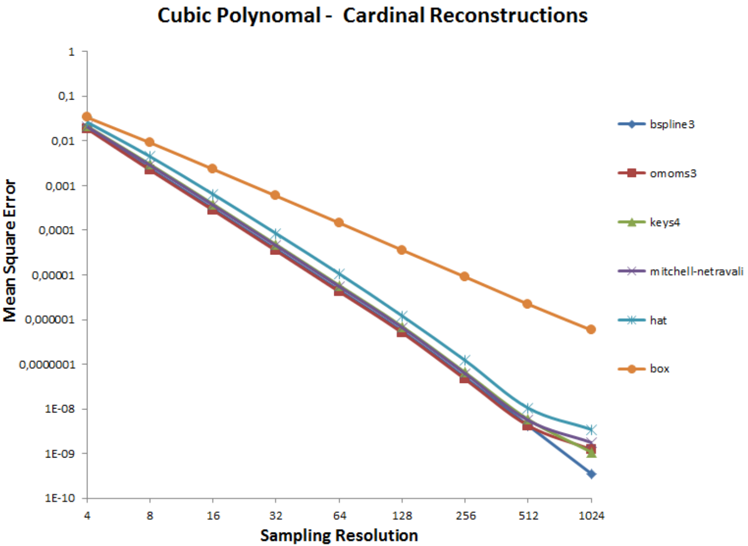
Experiment 3. Results 4



Experiment 4

Input. Cubic Polynomial: f
Output. Point Sampled Cardinal Reconstruction:
$\tilde{f} = [f] * [\varphi]^{-1} * \varphi$.
Objective. Identify the order of approximation that $\tilde{f}$ provides to f for some filters as a function of the sampling resolution.

Experiment 4. Results 1



Experiment 4. Results

Does **Bspline3** has the same order of polynomial approximation than **Hat**????...

Experiment 4. Results

Does **Bspline3** has the same order of polynomial approximation than **Hat**????...

NO!!!

Experiment 4. Results

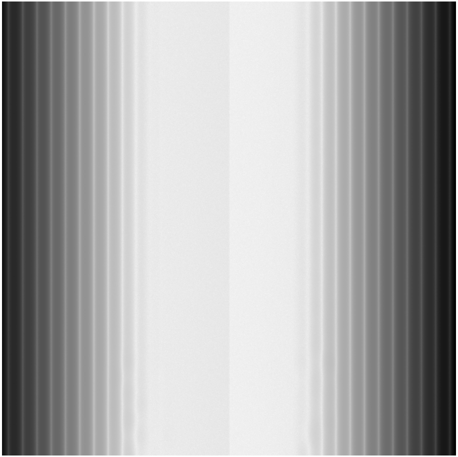
Does **Bspline3** has the same order of polynomial approximation than **Hat**????...

NO!!!

Where is the mistake???.....

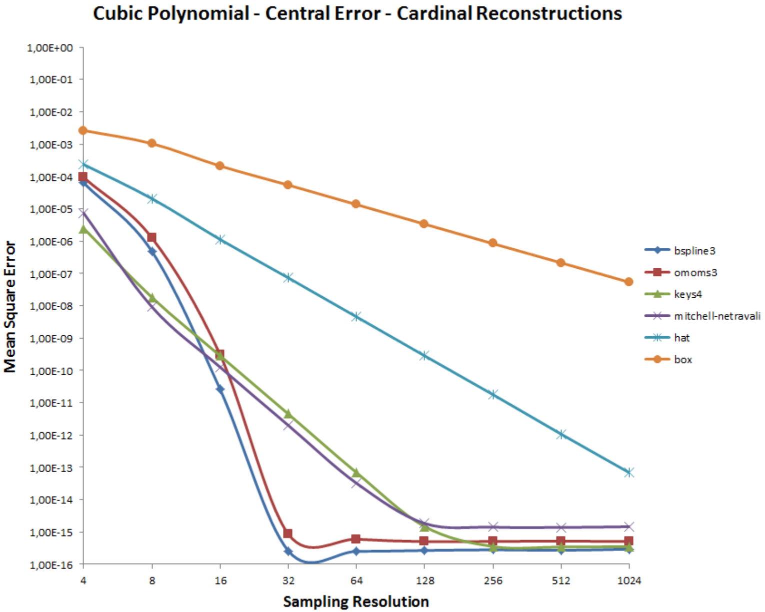
Experiment 4. Results 2 : Error Distribution Cardinal Bspline3

$p(x) = 4x^3 - 3x + 1$



Sampling Resolution: 32

Experiment 4. Results 3

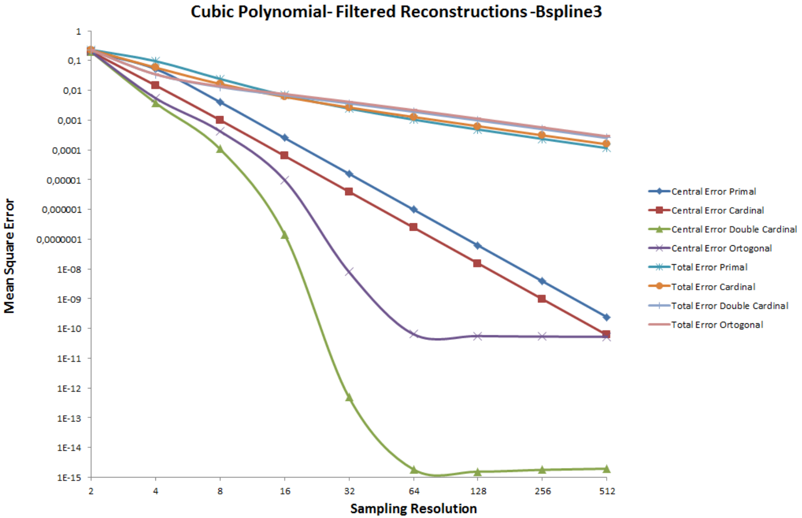


Experiment 5

Input. Cubic Polynomial: f
Output. Reconstructed Signal : $\tilde{f} = [f * \varphi] * d * \varphi$.
Objective. Identify the order of approximation that $\tilde{f}$ provides to f for a fixed type filter and some reconstruction methods.
Reconstruction Methods

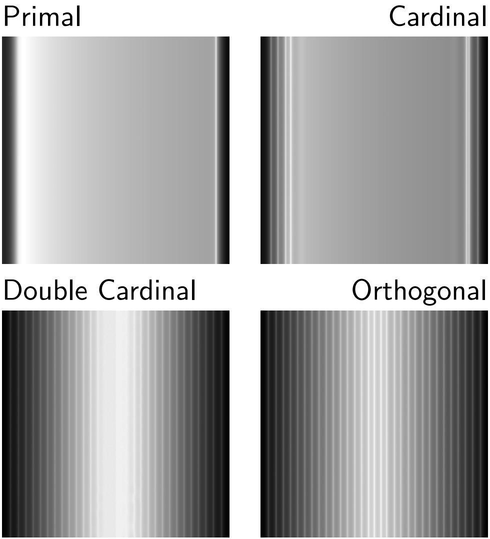
- Primal: $d = \delta$.
- Cardinal : $d = [\varphi]^{-1}$.
- Double Cardinal: $d = [\varphi]^{-1} * [\varphi]^{-1}$.
- Dual: $d = [\varphi * \varphi^{\vee}]^{-1}$

Experiment 5. Results 1



Experiment 5. Results 2: Error Distribution

$p(x) = 4x^3 - 3x + 1$



Sampling Resolution: 32

Conclusions

- We confirmed Bspline3 and Omoms3 have order of polynomal approximation 4, Mitchell and Keys have order 3, Hat order 2 and Box order 1.

Conclusions

- We confirmed Bspline3 and Omoms3 have order of polynomal approximation 4, Mitchell and Keys have order 3, Hat order 2 and Box order 1.
- We confirmed that any filter is unable to represent frequencies above Nyquist limit. Instead, such frequencies are transformed to noise.

Conclusions

- We confirmed Bspline3 and Omoms3 have order of polynomal approximation 4, Mitchell and Keys have order 3, Hat order 2 and Box order 1.
- We confirmed that any filter is unable to represent frequencies above Nyquist limit. Instead, such frequencies are transformed to noise.
- For a general texture, the error of approximation provided by the orthogonal projection in the filter space, is not visibly related with the order of polynomial approximation of the filter. Instead, the error of approximation (as a function of the sampling resolution) depends on the spectrum of the texture. Error just decrease as high frequencies go below Nyquist limit.

Obrigado!