MICRO-DOPPLER BASED RECOGNITION OF BALLISTIC TARGETS USING 2-D GABOR FILTERS

A.R. Persico, C. Clemente, C. V. Ilioudis, D. Gaglione,
J. Cao and J. J. Soraghan

Centre for Excellence in Signal and Image Processing, EEE, University of Strathclyde, Glasgow, UK

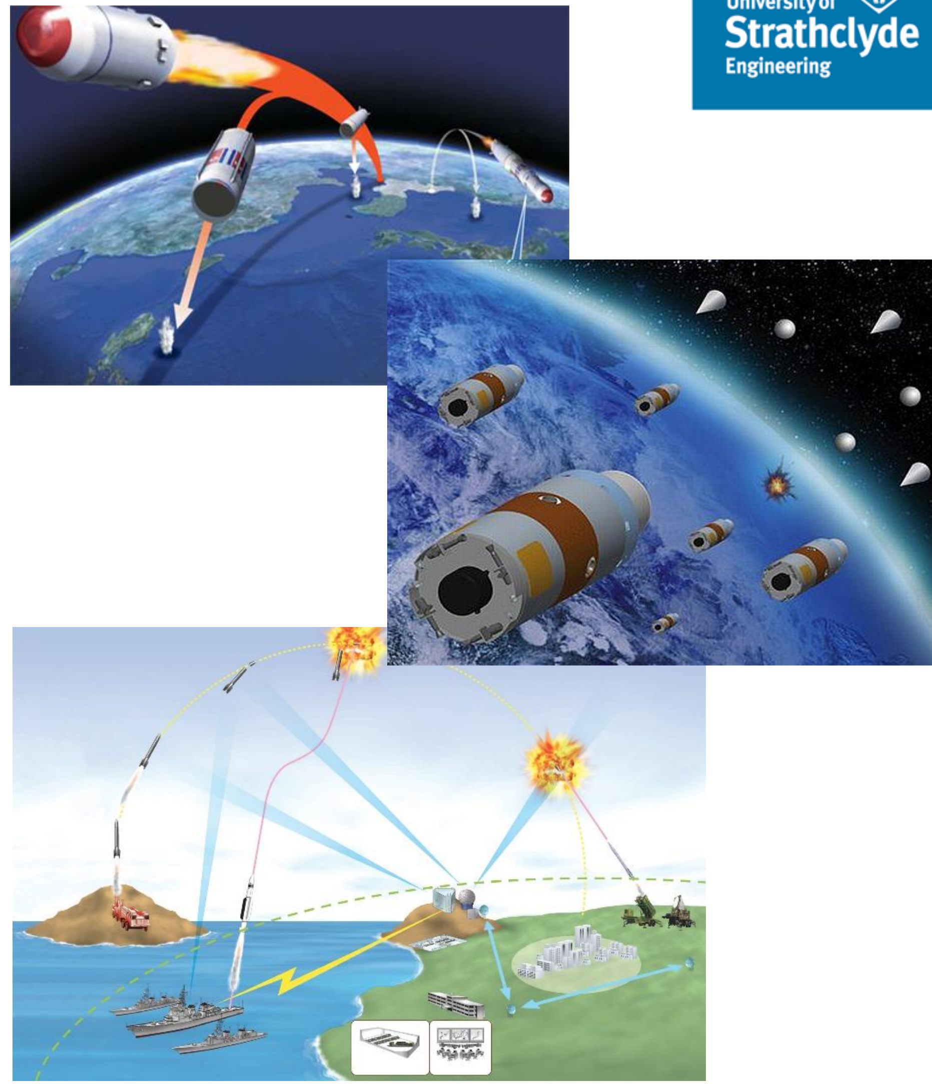
University of Strathclyde Engineering

CeSIP

Centre for excellence in Signal and Image Processing

OVERVIEW

- Introduction
  - Motivation
  - Aim
- 2-D Gabor Filter
- Algorithm Description
- Performance
- Conclusions



INTRODUCTION (1/2)

BALLISTIC MISSILE CLASSIFICATION FROM RADAR

- The capability to recognize **ballistic threats** is a critical topic due to the increasing effectiveness of the **warheads** and to **economical constraints**.

Introduction (1/2)

Ballistic Missile Classification from Radar

- The capability to recognize **ballistic threats** is a critical topic due to the increasing effectiveness of the **warheads** and to **economical constraints**.

- The ability to **distinguish** between **warheads** and **decoys** is crucial in order to mitigate the number of shots per hit and to maximize the ammunition capabilities.

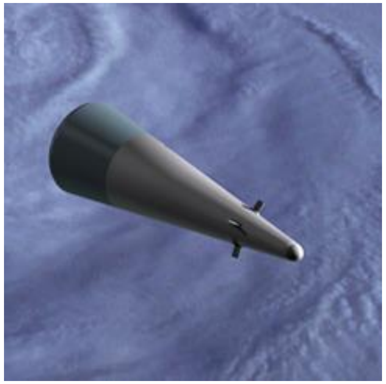
Introduction (1/2)

Ballistic Missile Classification from Radar

- The capability to recognize **ballistic threats** is a critical topic due to the increasing effectiveness of the **warheads** and to **economical constraints**.

- The ability to **distinguish** between **warheads** and **decoys** is crucial in order to mitigate the number of shots per hit and to maximize the ammunition capabilities.

  - Decoys comprises object of different shapes released by the missiles in order to introduce confusion in the interceptors.



warhead

Decoys

INTRODUCTION (2/2)

BALLISTIC MISSILE CLASSIFICATION FROM RADAR

- Since warheads and decoys exhibit different **micro-motions** during their ballistic trajectory, the **micro-Doppler effect analysis** is used to extract reliable information for target classification.

Introduction (2/2)

Ballistic Missile Classification from Radar

- Since warheads and decoys exhibit different **micro-motions** during their ballistic trajectory, the **micro-Doppler effect analysis** is used to extract reliable information for target classification.

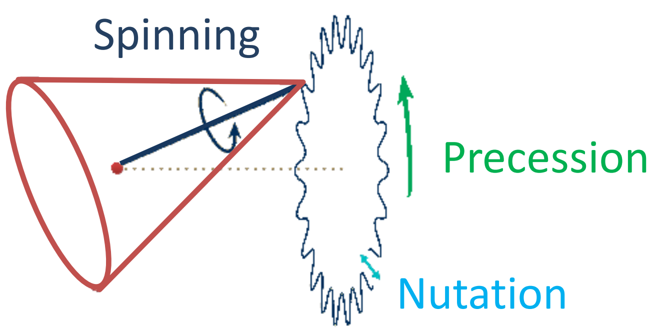
- The warheads are characterized by **precession** and **nutation**, while the decoys by **wobbling**.

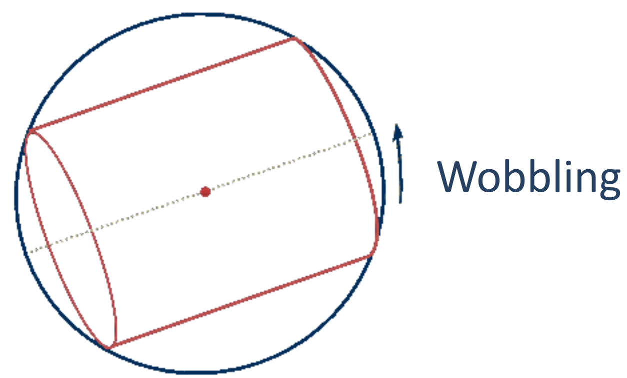
INTRODUCTION (2/2)

BALLISTIC MISSILE CLASSIFICATION FROM RADAR

- Since warheads and decoys exhibit different **micro-motions** during their ballistic trajectory, the **micro-Doppler effect analysis** is used to extract reliable information for target classification.

- The warheads are characterized by **precession** and **nutation**, while the decoys by **wobbling**.

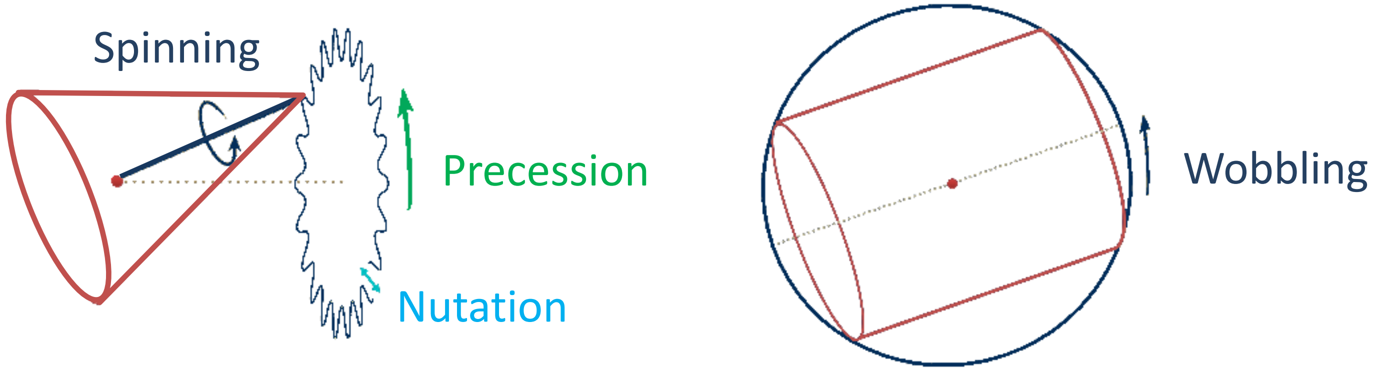




Introduction (2/2)
Ballistic Missile Classification from Radar

- Since warheads and decoys exhibit different **micro-motions** during their ballistic trajectory, the **micro-Doppler effect analysis** is used to extract reliable information for target classification.

- The warheads are characterized by **precession** and **nutation**, while the decoys by **wobbling**.



- Aim: Develop novel **classification algorithm** that is able to differentiate between **targets of interest** and **interference factors**, such as **decoys** and chaffs in an accurate and robust fashion.

2-D Gabor Filter (1/2)

Space and Harmonic Responses

The 2-D Gabor function is the product of a complex exponential representing a sinusoidal plane wave and an elliptical Gaussian in any rotation:

2-D Gabor Filter (1/2)

Space and Harmonic Responses

The 2-D Gabor function is the product of a complex exponential representing a sinusoidal plane wave and an elliptical Gaussian in any rotation:

- The **filter response** in the continuous domain can be normalized to have a compact closed form:

$$\psi(x,y) = \frac{f^2}{\pi\gamma\mu}\, e^{\left(\frac{f^2}{\gamma^2}x'^2 + \frac{f^2}{\mu^2}y'^2\right)}\, e^{j2\pi f x'}$$

$$x' = x\cos(\theta) + y\sin(\theta) \qquad y' = -\,x\sin(\theta) + y\cos(\theta)$$

where f is ***central spatial frequency*** of the filter, θ is the ***anticlockwise rotation*** of the Gaussian envelope and the sinusoidal plane wave, γ is the spatial width of the filter along the plane wave, and μ is the spatial width perpendicular to the wave.

2-D Gabor Filter (1/2)

Space and Harmonic Responses

The 2-D Gabor function is the product of a complex exponential representing a sinusoidal plane wave and an elliptical Gaussian in any rotation:

- The **filter response** in the continuous domain can be normalized to have a compact closed form:

$$\psi(x, y) = \frac{f^2}{\pi\gamma\mu}\, e^{\left(\frac{f^2}{\gamma^2}x'^2 + \frac{f^2}{\mu^2}y'^2\right)}\, e^{j2\pi f x'}$$

$$x' = x\cos(\theta) + y\sin(\theta) \qquad y' = -\,x\sin(\theta) + y\cos(\theta)$$

where f is ***central spatial frequency*** of the filter, θ is the ***anticlockwise rotation*** of the Gaussian envelope and the sinusoidal plane wave, γ is the spatial width of the filter along the plane wave, and μ is the spatial width perpendicular to the wave.

- The normalized filter **harmonic response** is:

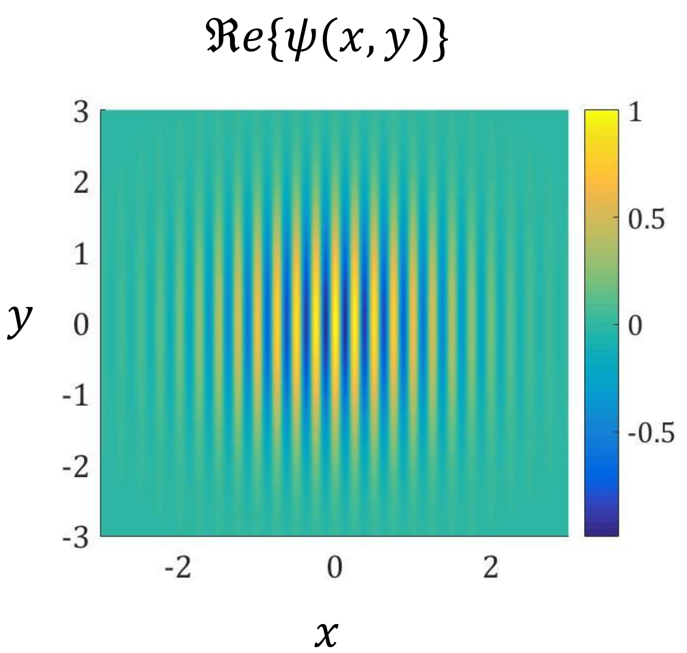
$$\Psi(u, v) = e^{-\frac{\pi^2}{f^2}\left(\gamma^2\left(u' - f\right)^2 + \mu^2 v'^2\right)}$$

$$u' = u\cos(\theta) + v\sin(\theta) \qquad v' = -\,u\sin(\theta) + v\cos(\theta)$$

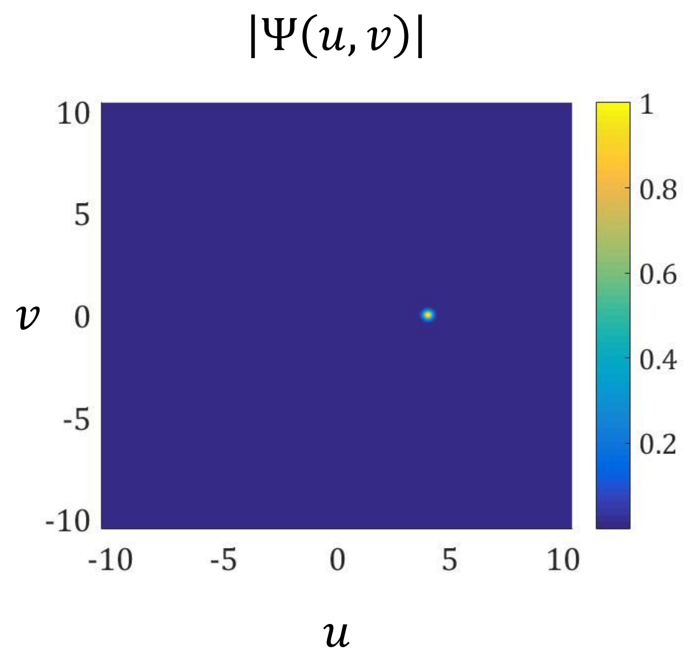
2-D Gabor Filter (2/2)

Varying Spatial Frequency and Orientation Angle

- Space Domain



- 2-D Fourier Domain



Parameters

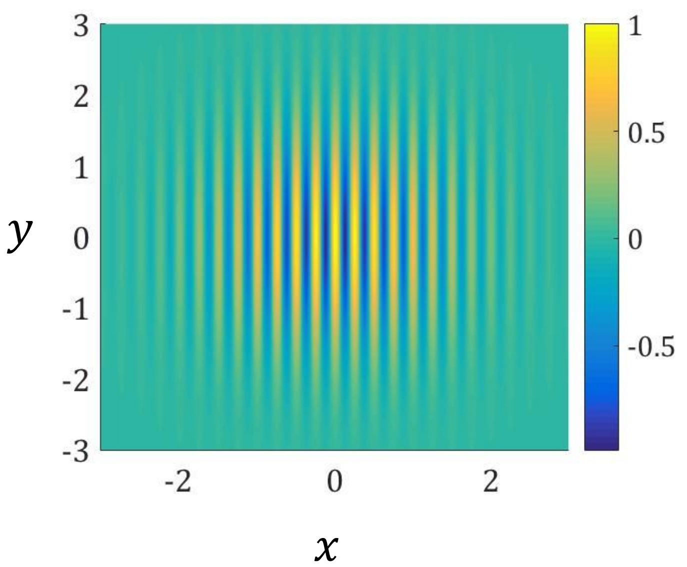
- $f = 4$
- $\theta = 0°$
- $\gamma = 2\pi$
- $\mu = 2\pi$

2-D Gabor Filter (2/2)

Varying Spatial Frequency and Orientation Angle

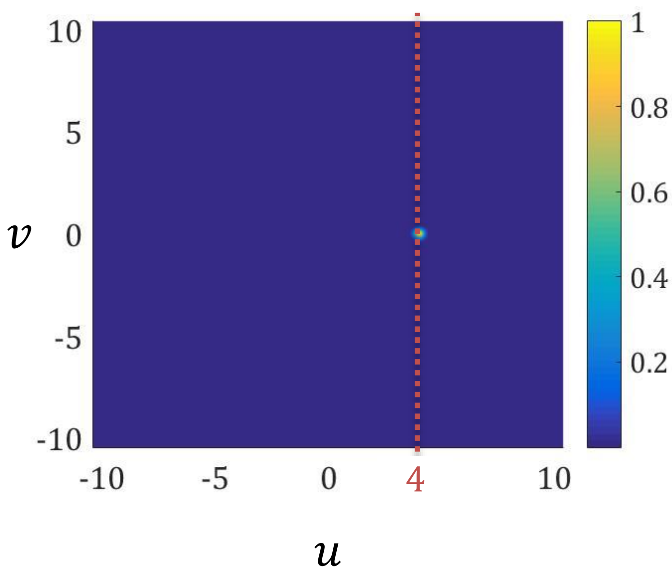
- Space Domain

$\Re e\{\psi(x,y)\}$



- 2-D Fourier Domain

$|\Psi(u,v)|$



Parameters

- $f = 4$
- $\theta = 0°$
- $\gamma = 2\pi$
- $\mu = 2\pi$

For $\theta = 0° \rightarrow u' = u\,;\; v' = v$.

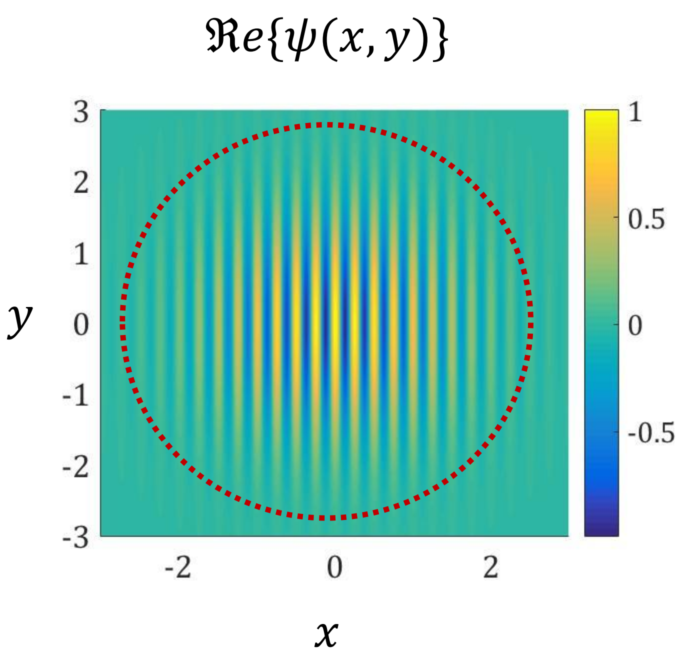
$$\Psi(u,v) = e^{-\frac{\pi^2}{f^2}\left(\gamma^2(u-f)^2+\mu^2 v^2\right)}$$

Therefore the harmonic response is centred in $\mathrm{u} = f$.

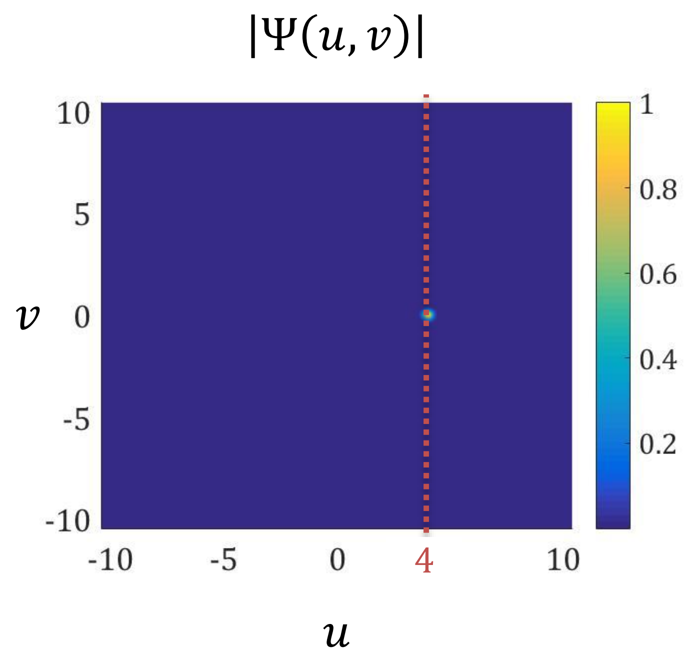
2-D Gabor Filter (2/2)

Varying Spatial Frequency and Orientation Angle

- Space Domain



- 2-D Fourier Domain



Parameters

- $f = 4$
- $\theta = 0°$
- $\gamma = 2\pi$
- $\mu = 2\pi$

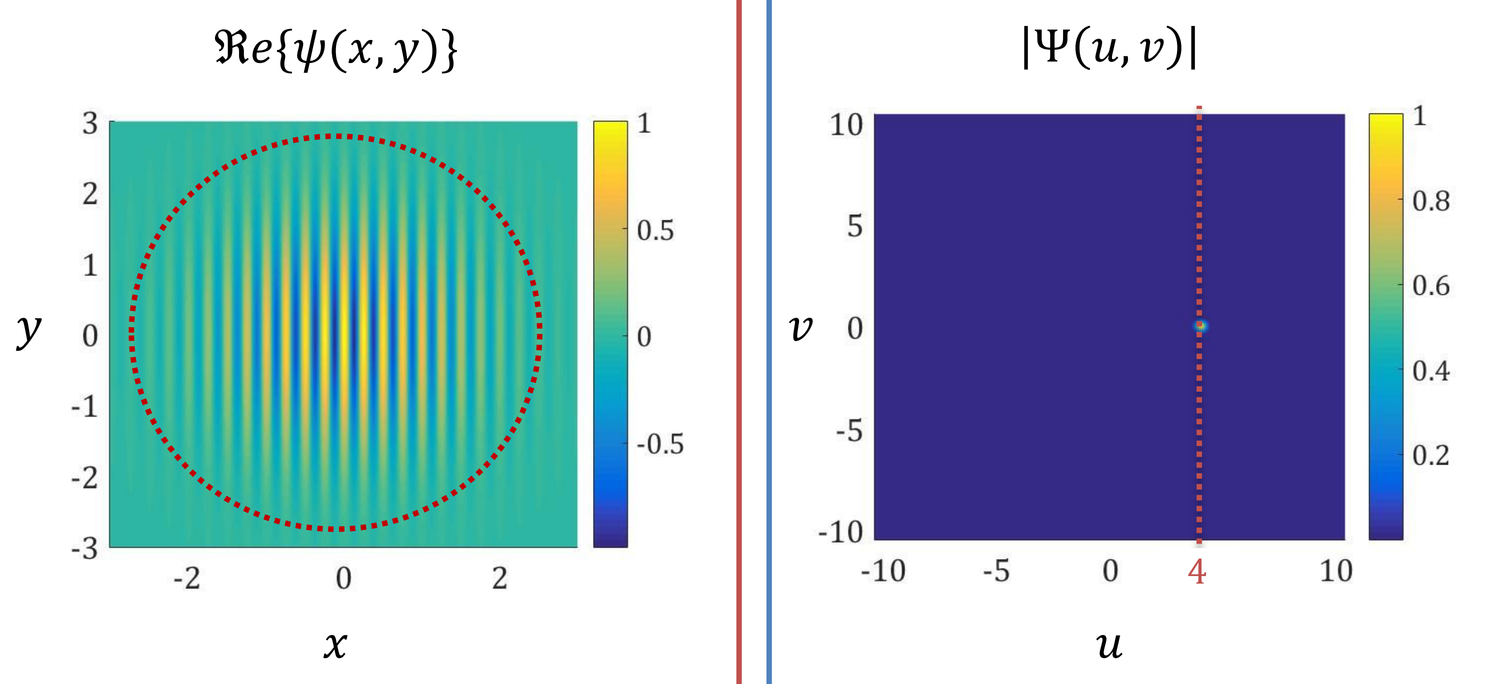
The Gaussian envelope is circular due to $\mu = \gamma$

2-D Gabor Filter (2/2)

Varying Spatial Frequency and Orientation Angle

- Space Domain

- 2-D Fourier Domain



Parameters

- $f = 4$
- $\theta = 0°$
- $\gamma = 2\pi$
- $\mu = 2\pi$

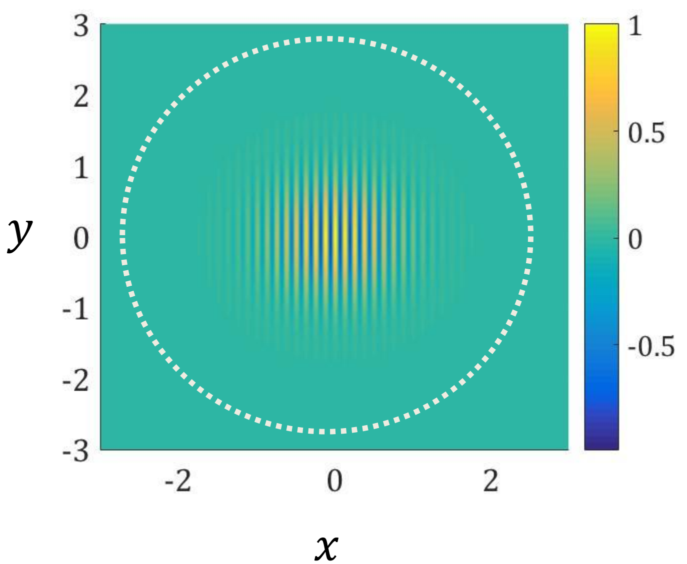
Changing central spatial frequency

2-D Gabor Filter (2/2)

Varying Spatial Frequency and Orientation Angle

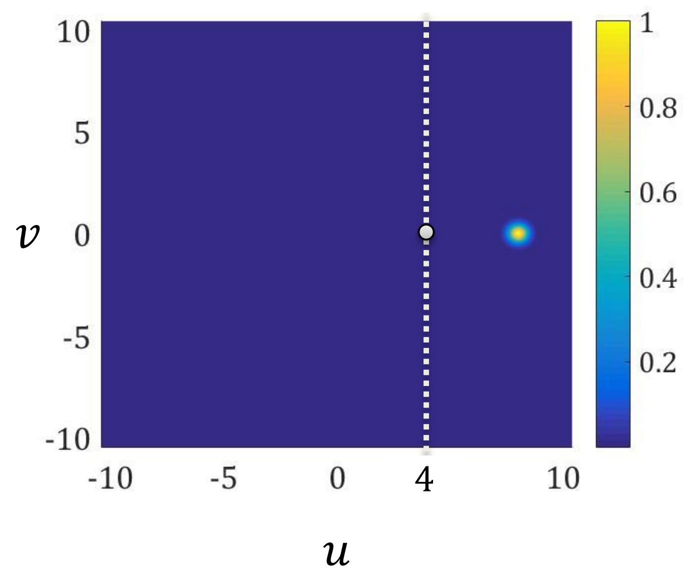
- Space Domain

$\Re e\{\psi(x,y)\}$



- 2-D Fourier Domain

$|\Psi(u,v)|$



Parameters

- $f = 8$
- $\theta = 0°$
- $\gamma = 2\pi$
- $\mu = 2\pi$

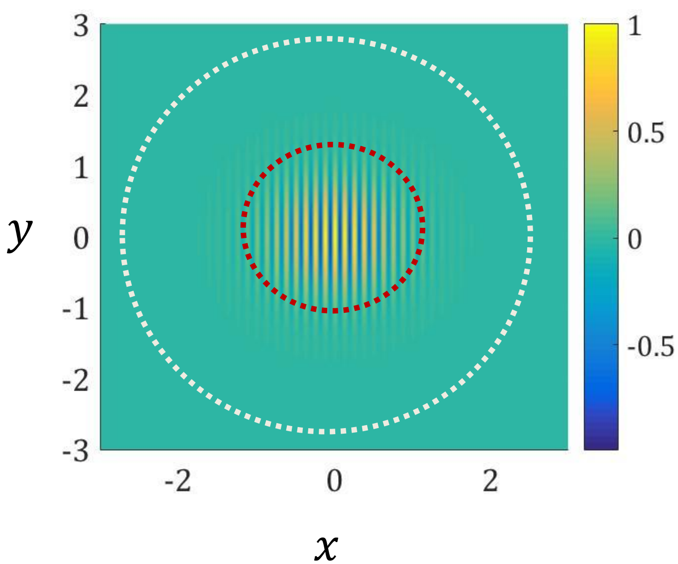
Changing central spatial frequency

2-D Gabor Filter (2/2)

Varying Spatial Frequency and Orientation Angle

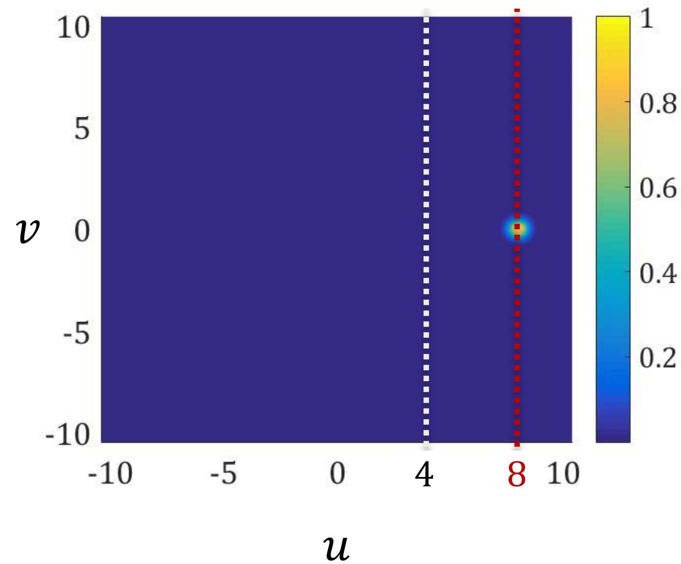
- Space Domain

$\Re e\{\psi(x,y)\}$



- 2-D Fourier Domain

$|\Psi(u,v)|$



Parameters

- $f = 8$
- $\theta = 0°$
- $\gamma = 2\pi$
- $\mu = 2\pi$

Changing central spatial frequency

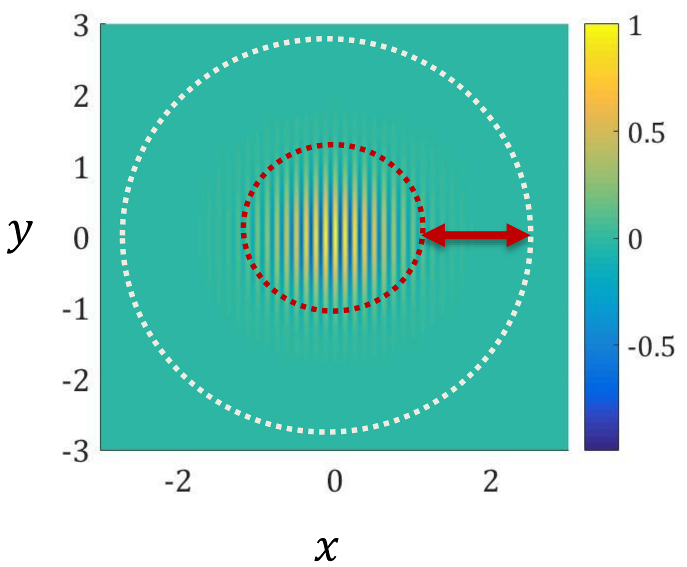
- The harmonic response is shifted along the u-axis
- The response width along both the axis decreases as f increases

2-D Gabor Filter (2/2)

Varying Spatial Frequency and Orientation Angle

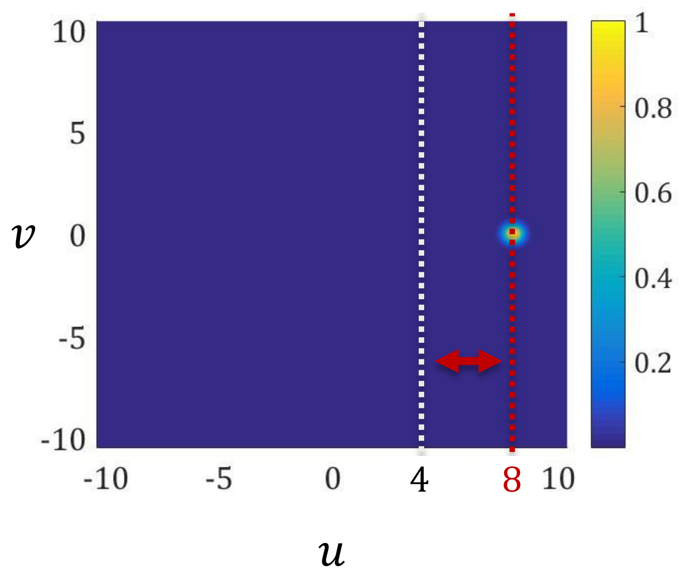
- Space Domain

$\Re e\{\psi(x,y)\}$



- 2-D Fourier Domain

$|\Psi(u,v)|$



Parameters

- $f = 8$
- $\theta = 0°$
- $\gamma = 2\pi$
- $\mu = 2\pi$

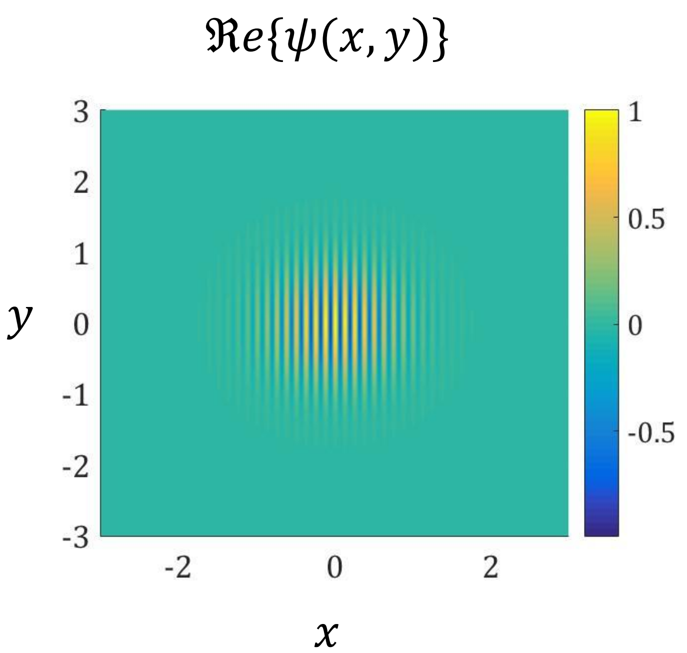
Changing central spatial frequency

- The harmonic response is shifted along the u-axis
- The response width along both the axis decreases as f increases

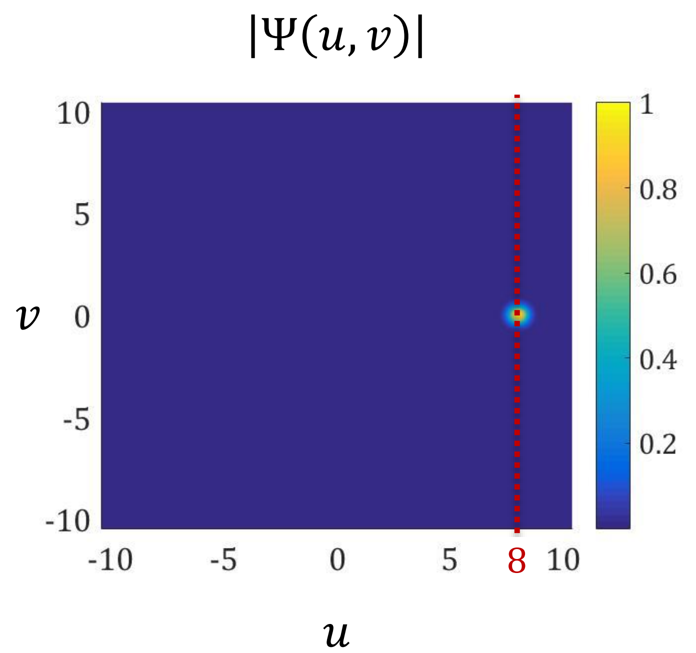
2-D Gabor Filter (2/2)

Varying Spatial Frequency and Orientation Angle

- Space Domain



- 2-D Fourier Domain



Parameters

- $f = 8$
- $\boxed{\theta = 0^\circ}$
- $\gamma = 2\pi$
- $\mu = 2\pi$

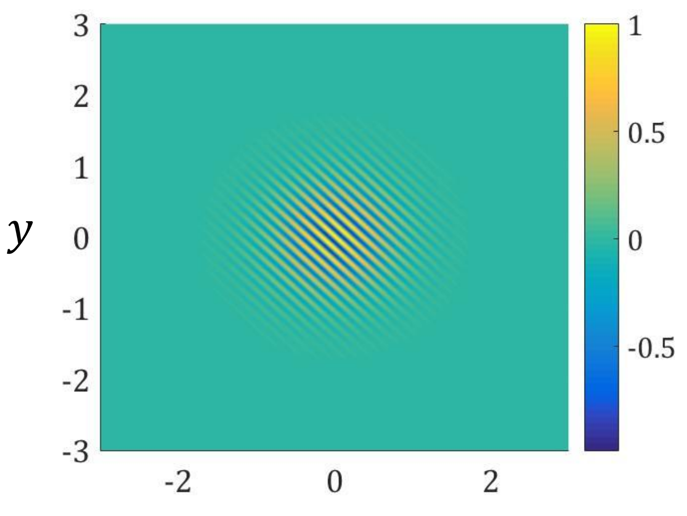
Changing orientation angle

2-D Gabor Filter (2/2)

Varying Spatial Frequency and Orientation Angle

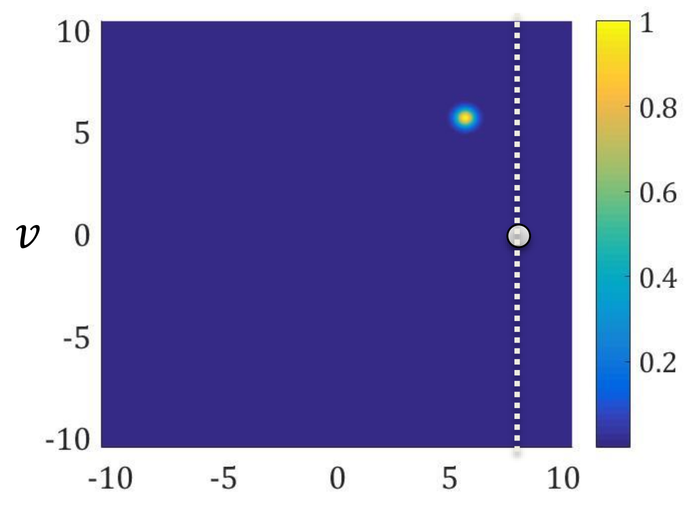
- Space Domain

$\Re e\{\psi(x,y)\}$



- 2-D Fourier Domain

$|\Psi(u,v)|$



Parameters

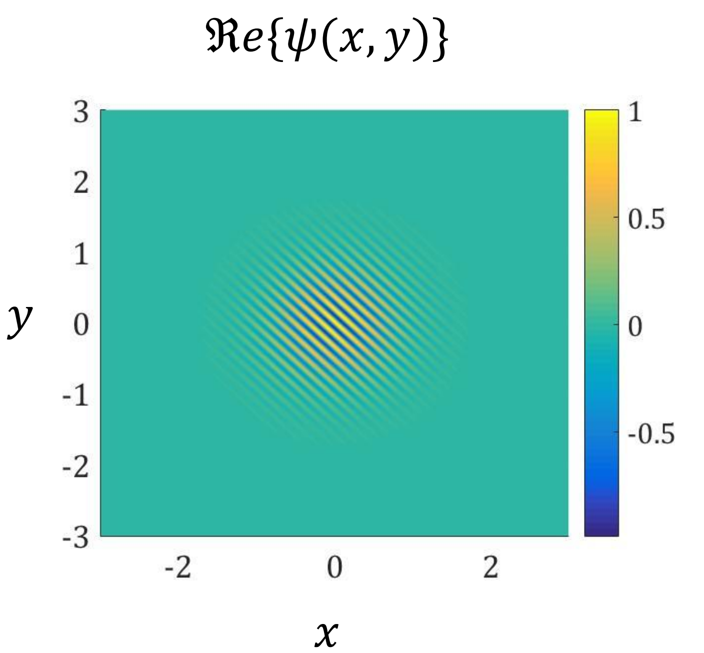
- $f = 8$
- $\theta = 45°$
- $\gamma = 2\pi$
- $\mu = 2\pi$

Changing orientation angle

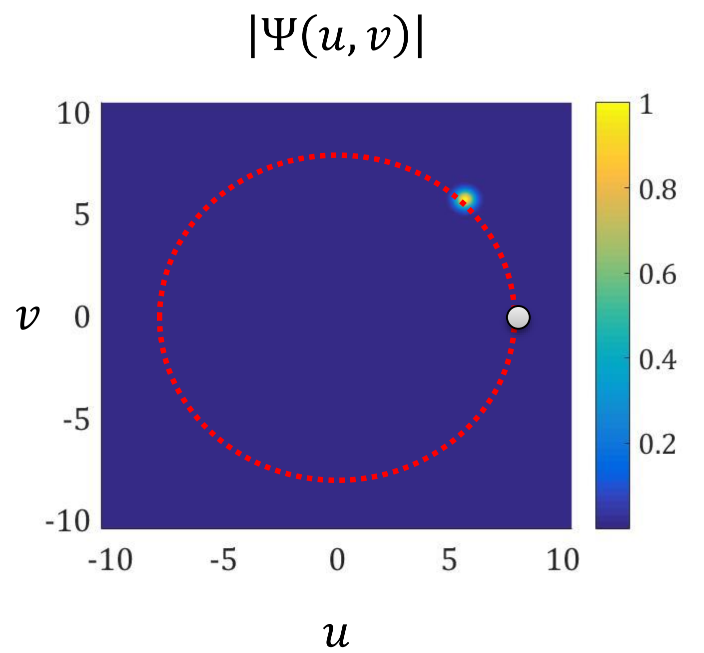
2-D Gabor Filter (2/2)

Varying Spatial Frequency and Orientation Angle

- Space Domain



- 2-D Fourier Domain



Parameters

- $f = 8$
- $\theta = 45°$
- $\gamma = 2\pi$
- $\mu = 2\pi$

Changing orientation angle

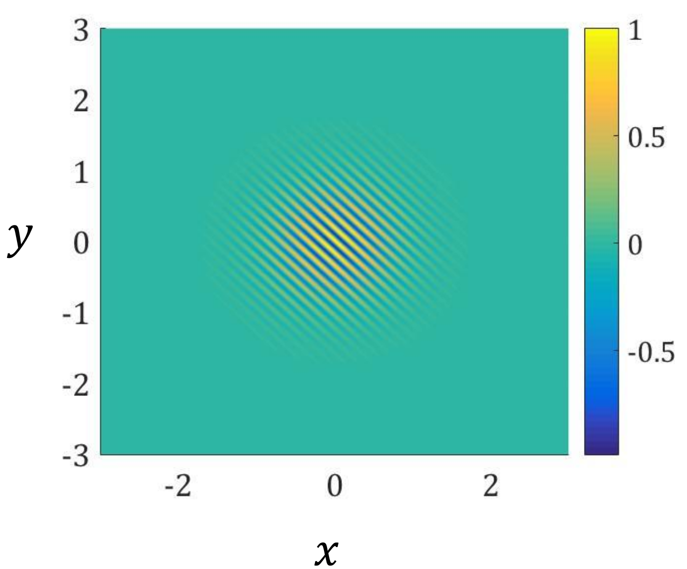
The harmonic response moves anticlockwise of an angle θ on a circumference whose radius is equal to f

2-D Gabor Filter (2/2)

Varying Spatial Frequency and Orientation Angle

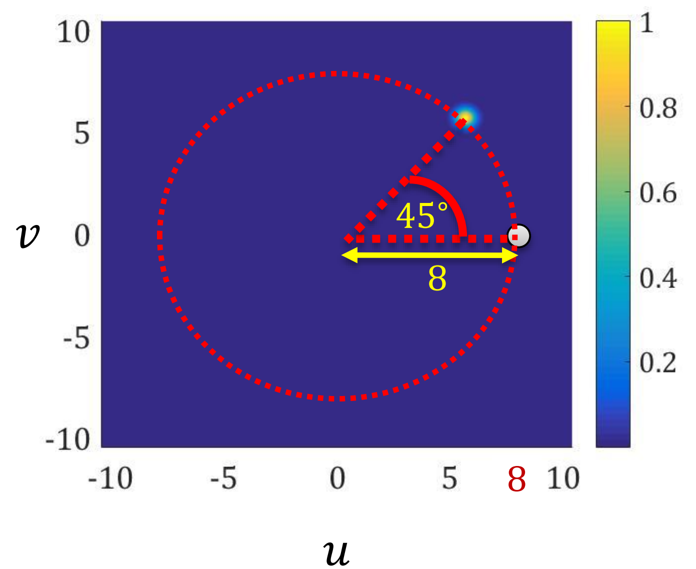
- Space Domain

$\Re e\{\psi(x,y)\}$



- 2-D Fourier Domain

$|\Psi(u,v)|$



Parameters

- $f = 8$
- $\theta = 45°$
- $\gamma = 2\pi$
- $\mu = 2\pi$

Changing orientation angle

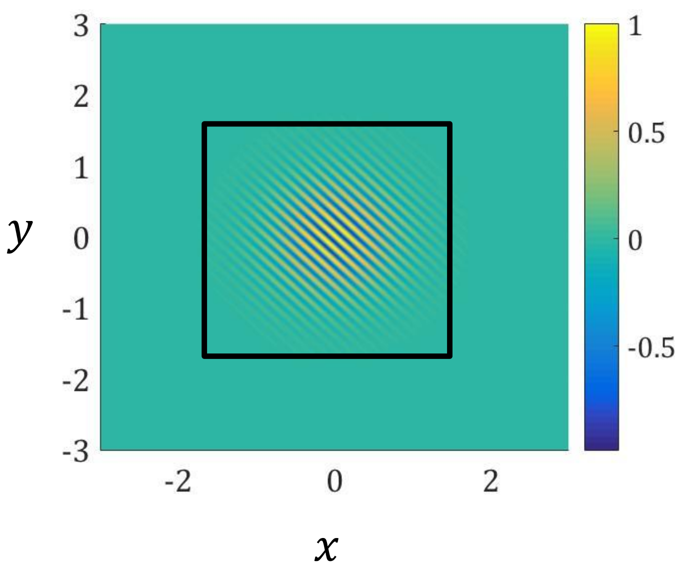
The harmonic response moves anticlockwise of an angle θ on a circumference whose radius is equal to f

2-D Gabor Filter (2/2)

Varying Spatial Frequency and Orientation Angle

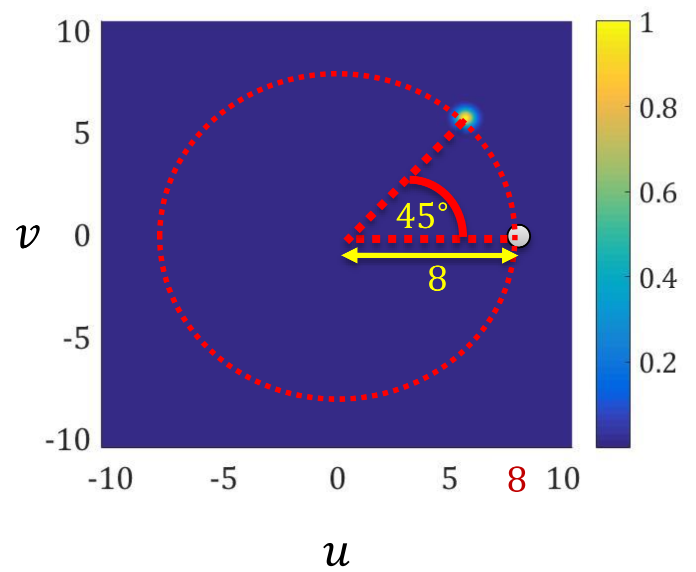
- Space Domain

$\Re e\{\psi(x,y)\}$



- 2-D Fourier Domain

$|\Psi(u,v)|$



Parameters

- $f = 8$
- $\theta = 45°$
- $\gamma = 2\pi$
- $\mu = 2\pi$

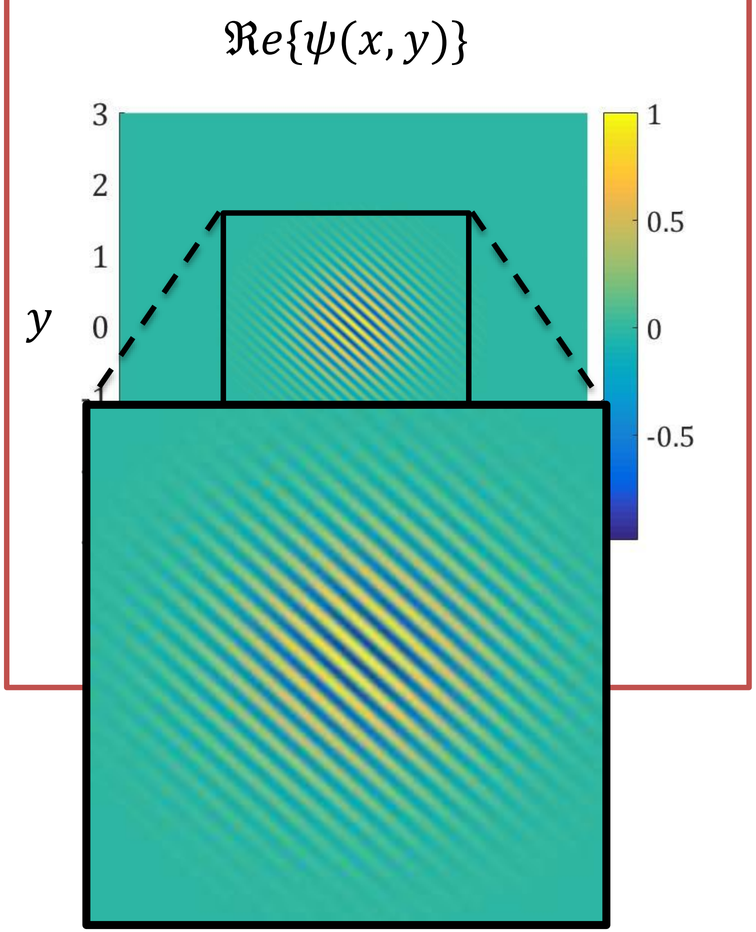
Changing orientation angle

The harmonic response moves anticlockwise of an angle θ on a circumference whose radius is equal to f

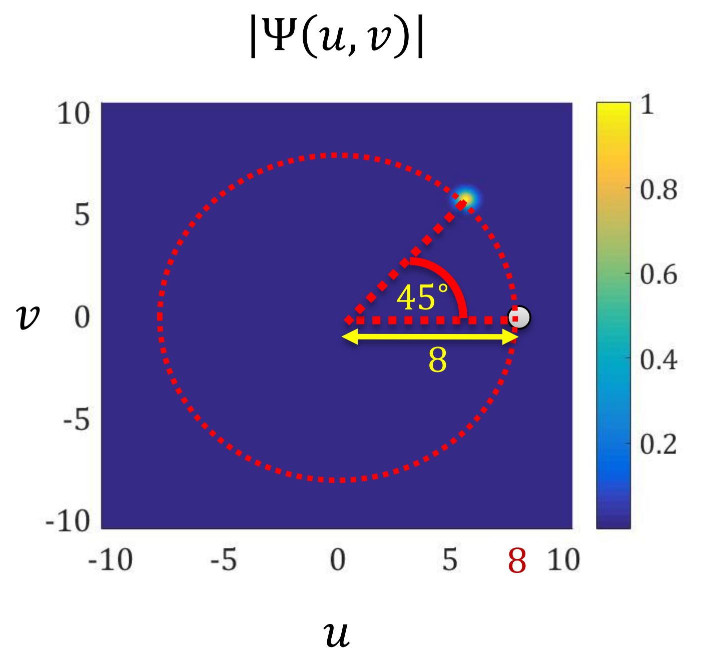
2-D Gabor Filter (2/2)

Varying Spatial Frequency and Orientation Angle

- Space Domain



- 2-D Fourier Domain



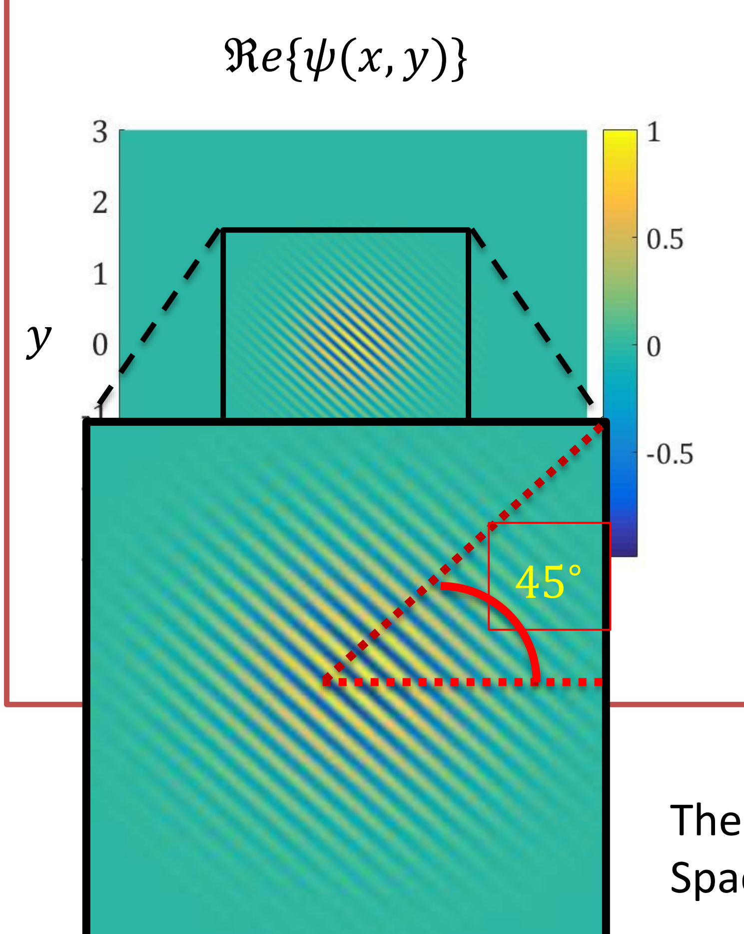
Parameters

- $f = 8$
- $\theta = 45°$
- $\gamma = 2\pi$
- $\mu = 2\pi$

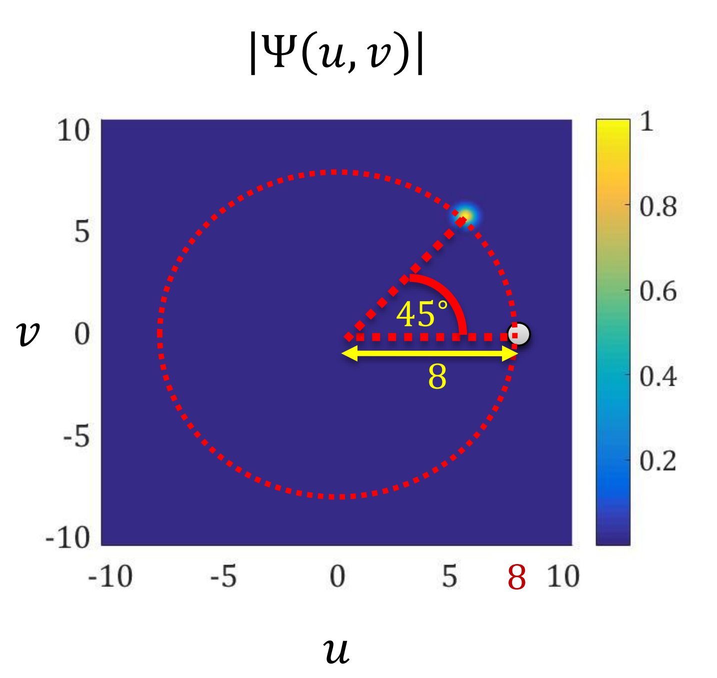
2-D Gabor Filter (2/2)

Varying Spatial Frequency and Orientation Angle

- Space Domain



- 2-D Fourier Domain



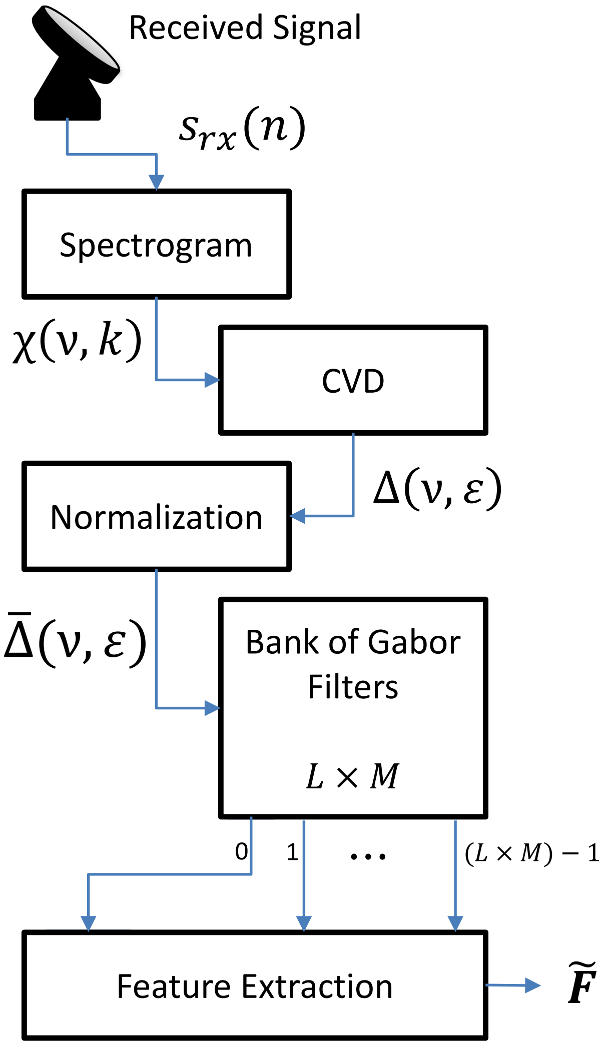
Parameters

- $f = 8$
- $\theta = 45°$
- $\gamma = 2\pi$
- $\mu = 2\pi$

The filter response is rotated of an angle θ also in the Space Domain

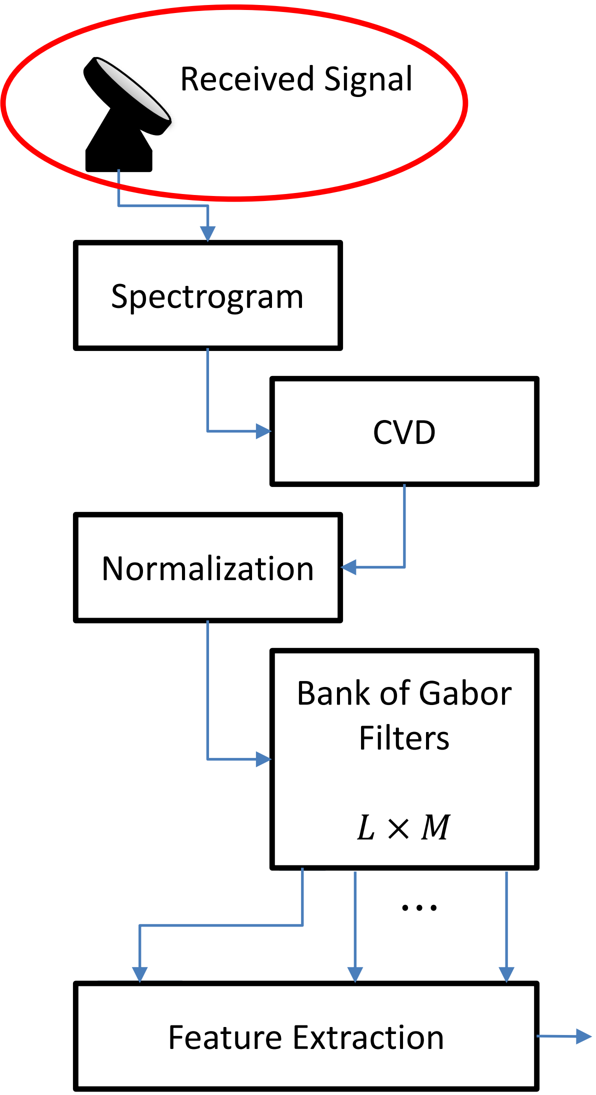
ALGORITHM (1/6)

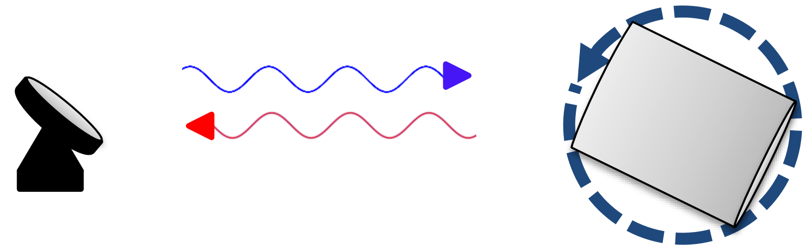
BLOCK SCHEME



Algorithm (2/6)

Received Signal

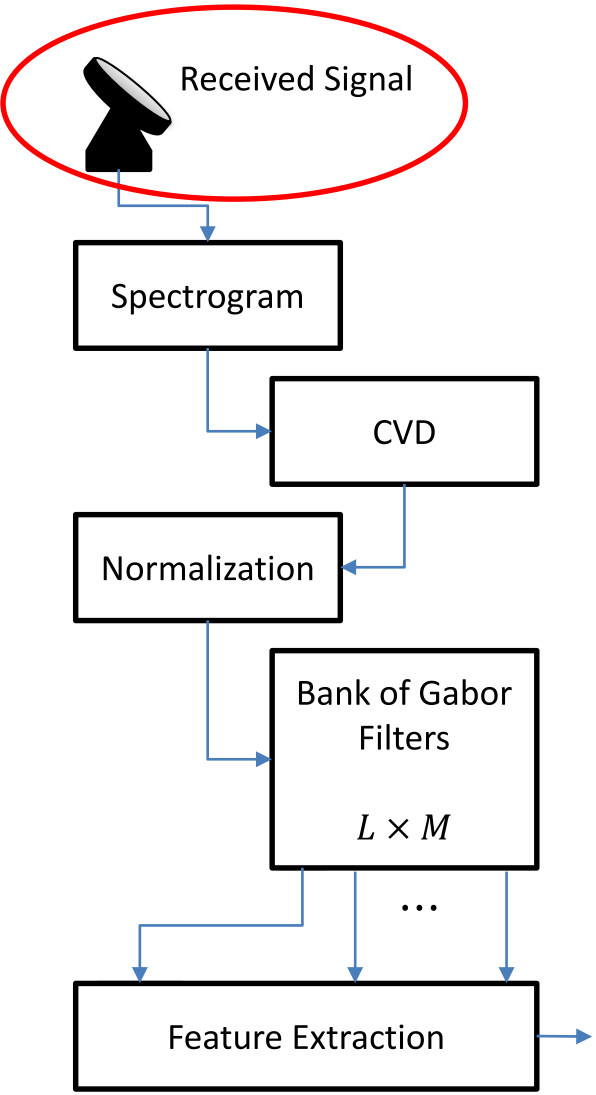


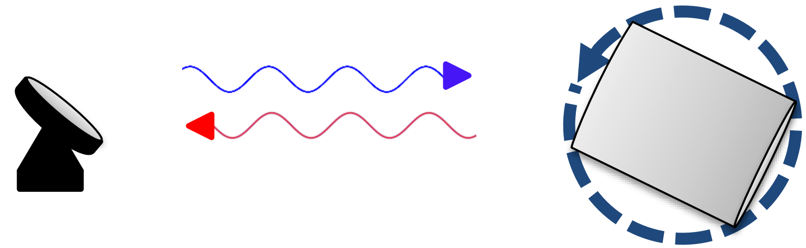


$$S_{rx}(n) = signal(n) / \max_{n} signal\,(n)$$

Algorithm (2/6)

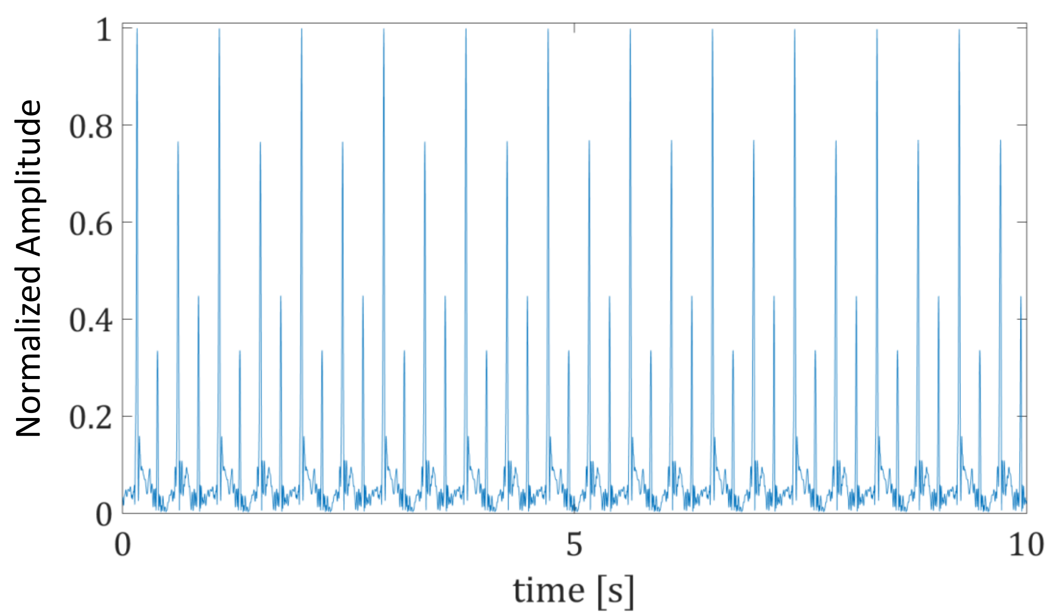
Received Signal





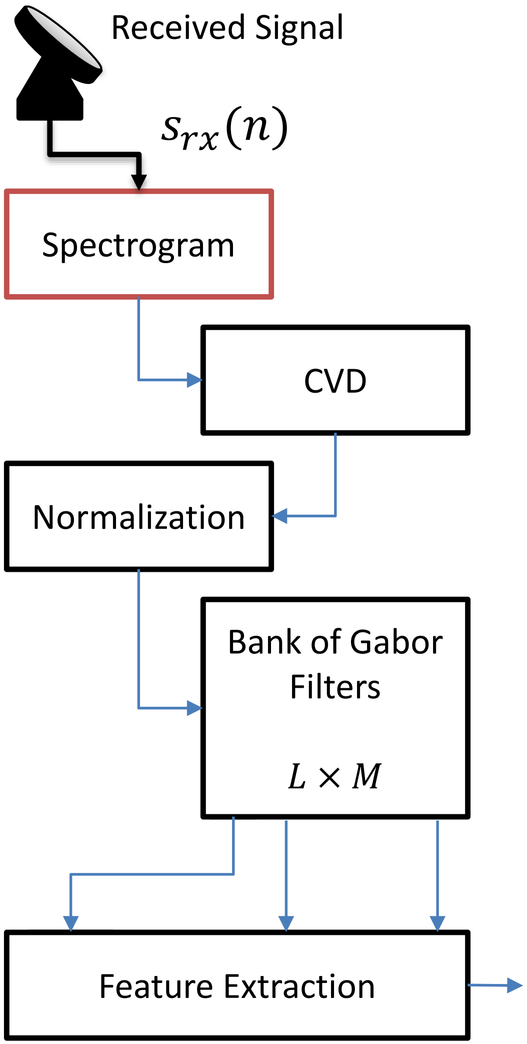
$$S_{rx}(n) = signal(n) / \max_{n} signal\,(n)$$

The example shows a signal from a wobbling cylinder.



ALGORITHM (3/6)

SPECTROGRAM



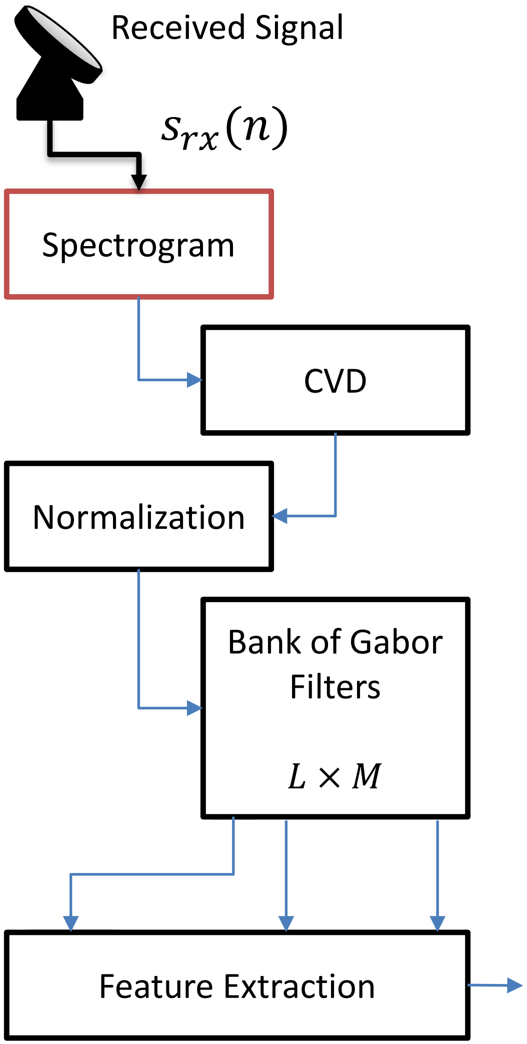
- The **Spectrogram** is the modulus of the **STFT** (**S**hort **T**ime **F**ourier **T**ransform) of the received signal

$$\chi(\nu, k) = \left| \sum_{n=0}^{N-1} s_{rx}(n) w_h(n-k) \exp\left(-j2\nu \frac{n}{N}\right) \right|$$

Where ν is the normalized frequency and $w_h(.)$ is the smoothing window.

ALGORITHM (3/6)

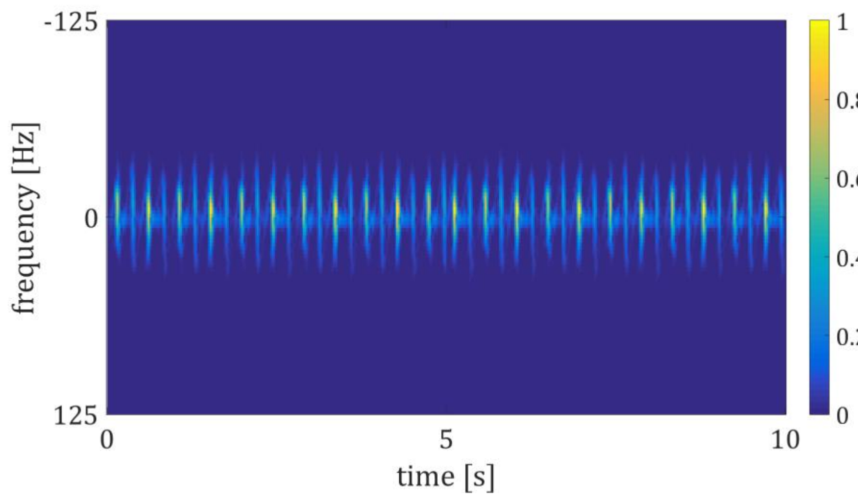
SPECTROGRAM



- The **Spectrogram** is the modulus of the **STFT** (**S**hort **T**ime **F**ourier **T**ransform) of the received signal

$$\chi(\nu, k) = \left| \sum_{n=0}^{N-1} s_{rx}(n) w_h(n-k) \exp\left(-j2\nu \frac{n}{N}\right) \right|$$

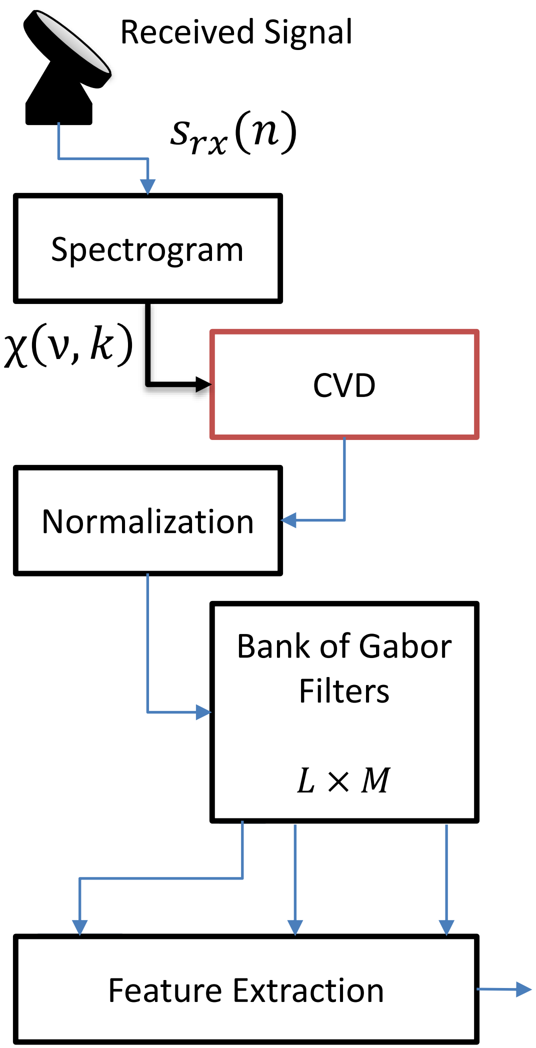
Where ν is the normalized frequency and $w_h(.)$ is the smoothing window.



It allows to evaluate the signal frequency variation on time

ALGORITHM (4/6)

CADENCE VELOCITY DIAGRAM



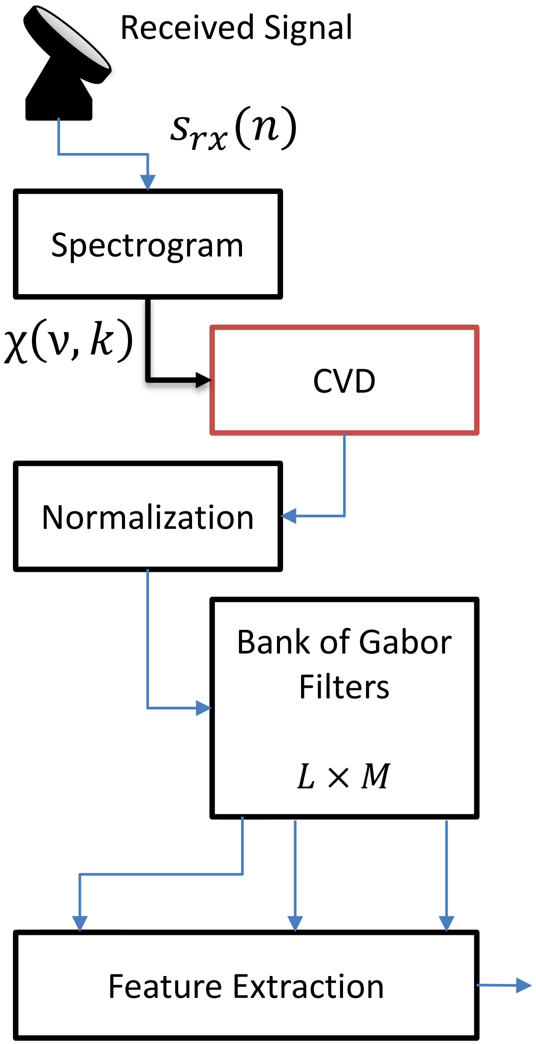
- The **CVD** (Cadence Velocity Diagram) [1], is defined as the modulus of the Fourier Transform of the Spectrogram along each frequency bin

$$\Delta(\nu, \varepsilon) = \left| \sum_{n=0}^{K-1} \chi(\nu, k) \exp\left(-j2\nu \frac{k}{K}\right) \right|$$

[1] C. Clemente, L. Pallotta, A. De Maio, J. Soraghan, and A. Farina, "A novel algorithm for radar classification based on Doppler characteristics exploiting orthogonal pseudo-Zernike polynomials," IEEE Transactions on Aerospace and Electronic Systems, vol. 51, no. 1, pp. 417–430, 2015.

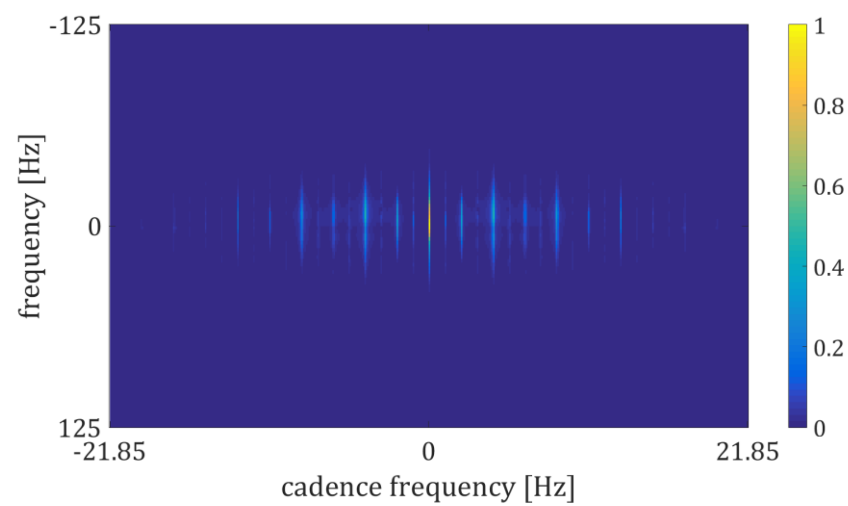
Algorithm (4/6)

Cadence Velocity Diagram



- The **CVD** (Cadence Velocity Diagram) [1], is defined as the modulus of the Fourier Transform of the Spectrogram along each frequency bin

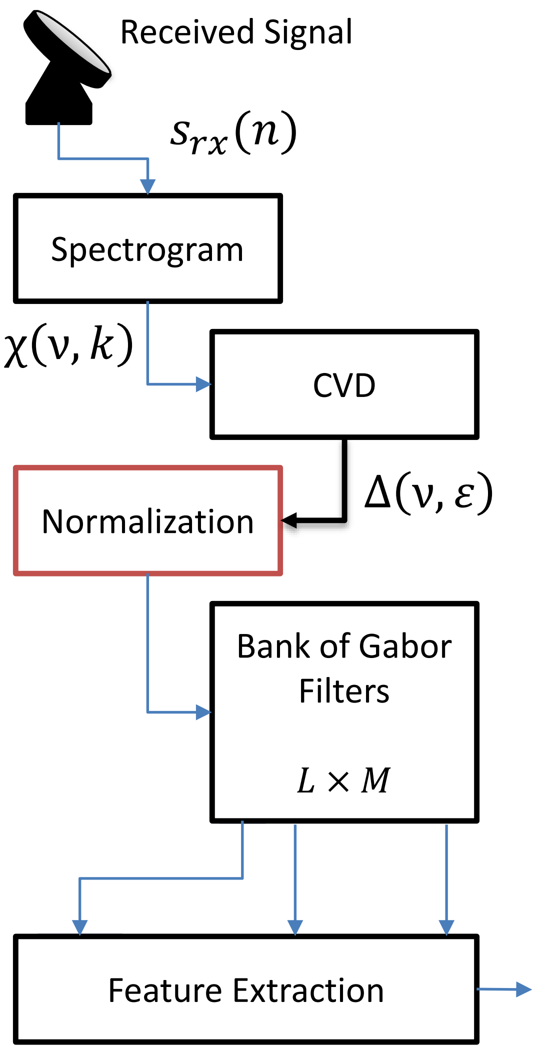
$$\Delta(\nu, \varepsilon) = \left| \sum_{n=0}^{K-1} \chi(\nu, k) \exp\left(-j2\nu \frac{k}{K}\right) \right|$$



[1] C. Clemente, L. Pallotta, A. De Maio, J. Soraghan, and A. Farina, "A novel algorithm for radar classification based on Doppler characteristics exploiting orthogonal pseudo-Zernike polynomials," IEEE Transactions on Aerospace and Electronic Systems, vol. 51, no. 1, pp. 417–430, 2015.

Algorithm (5/6)

Normalization and Filtering



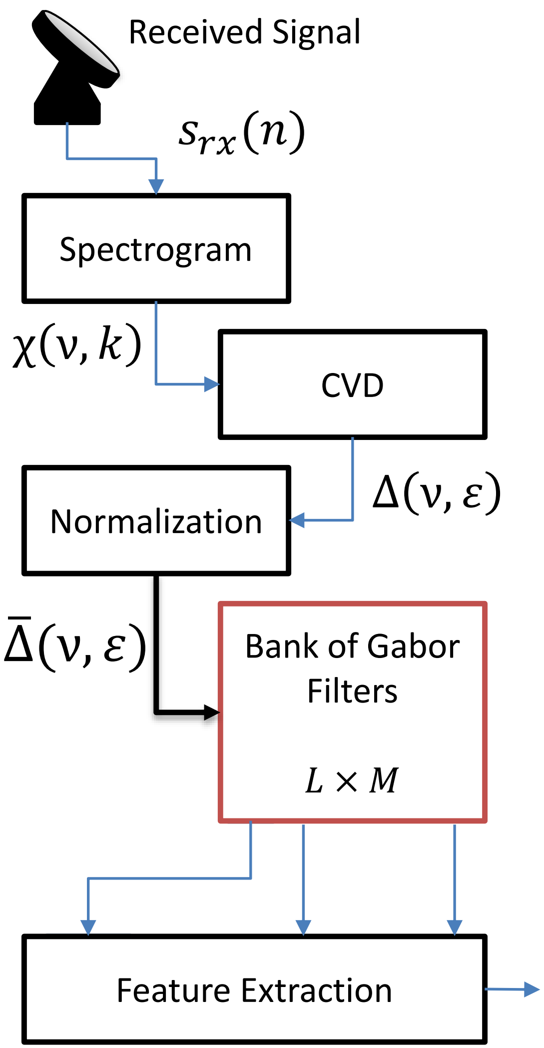
- The **CVD** is **normalized** in order to obtain a matrix whose values lie in the range $[\mathbf{0}, \mathbf{1}]$

$$\bar{\Delta}(\nu, \epsilon) = \frac{\Delta(\nu, \epsilon) - \min_{\nu,\varepsilon} \Delta(\nu, \epsilon)}{\max_{\nu,\varepsilon}[\Delta(\nu, \epsilon) - \min_{\nu,\varepsilon} \Delta(\nu, \epsilon)]}$$

Whose each element is considered as a pixel of a **2D-image**.

Algorithm (5/6)

Normalization and Filtering



- The **CVD** is **normalized** in order to obtain a matrix whose values lie in the range $[\mathbf{0}, \mathbf{1}]$

$$\bar{\Delta}(\nu, \epsilon) = \frac{\Delta(\nu, \epsilon) - \min\limits_{\nu,\varepsilon} \Delta(\nu, \epsilon)}{\max\limits_{\nu,\varepsilon}[\Delta(\nu, \epsilon) - \min\limits_{\nu,\varepsilon} \Delta(\nu, \epsilon)]}$$

  Whose each element is considered as a pixel of a **2D-image**.

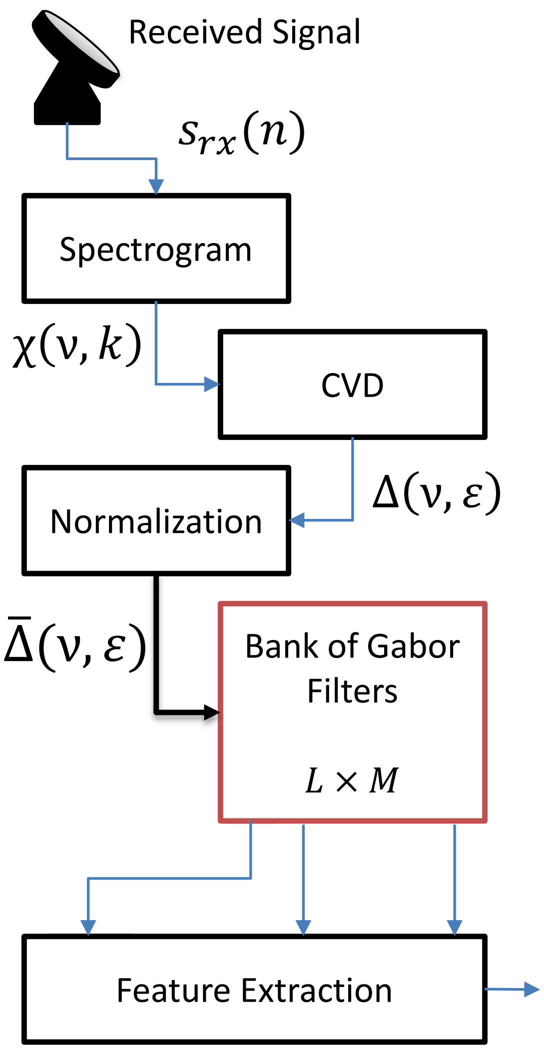
- The image is given as input to a **bank of Gabor filters** on varying the orientation angle and the central frequency.

  The output image is given by the **convolution product** of the Gabor function and the input image

$$g(\nu, \varepsilon; f_l, \theta_m) = \psi_{l,m}(\nu, \varepsilon; f_l, \theta_m) * \bar{\Delta}(\nu, \varepsilon)$$

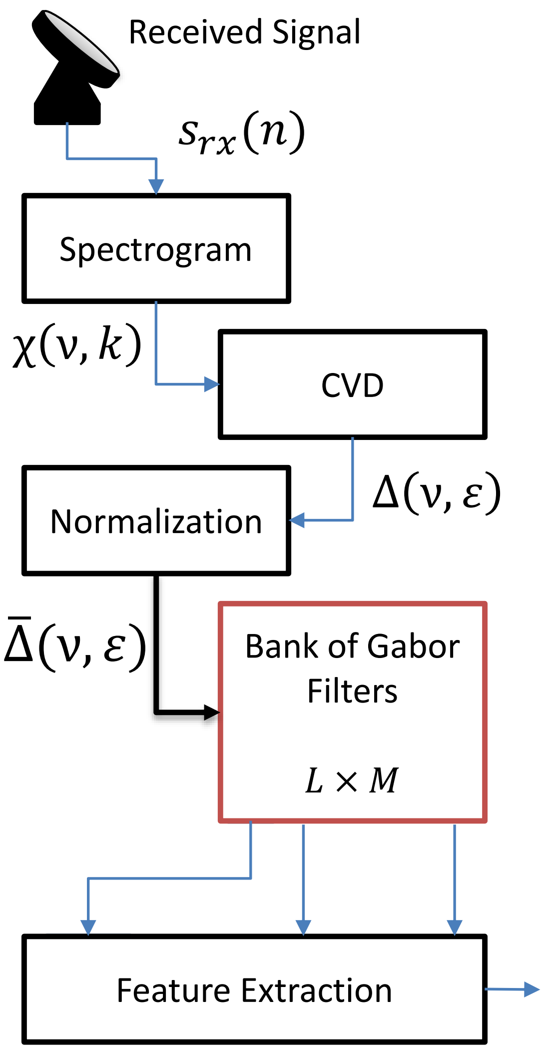
ALGORITHM (5/6)

NORMALIZATION AND FILTERING

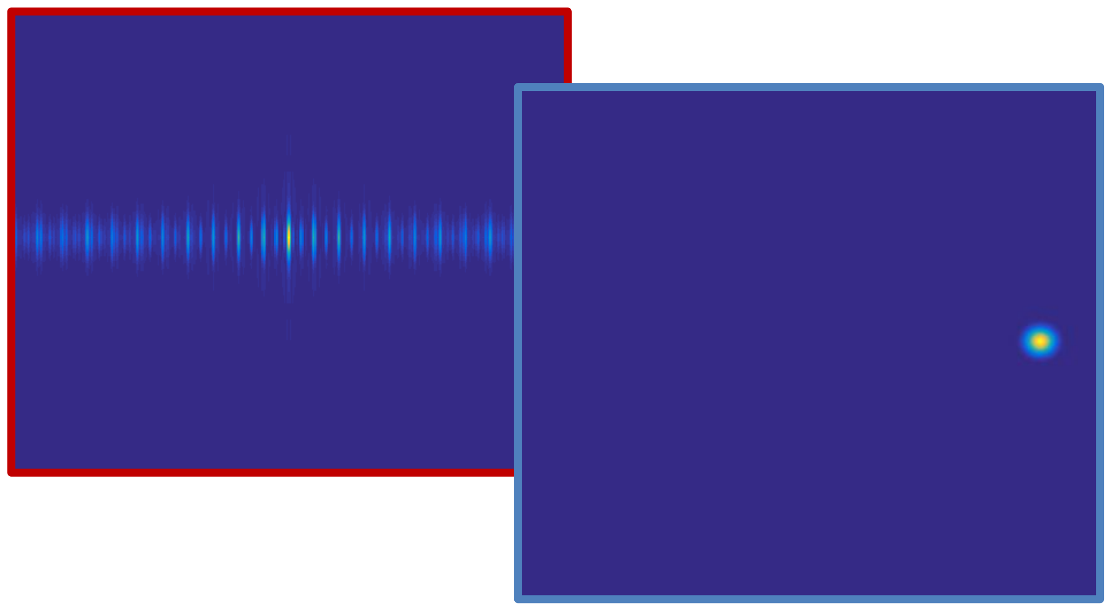


- The convolution product can be made in the Fourier Domain as the product of the **transformed input** and the **harmonic responses** of the filter

Algorithm (5/6)
Normalization and Filtering



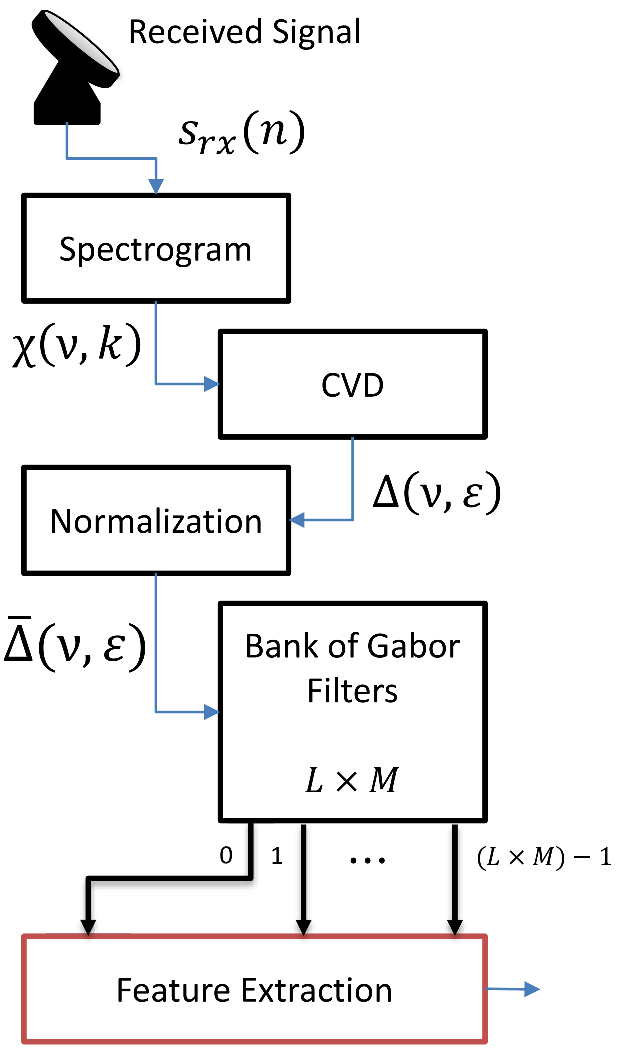
- The convolution product can be made in the Fourier Domain as the product of the **transformed input** and the **harmonic responses** of the filter



Since both the **CVD** and its **2D Fourier transform** are characterized by **vertical lines** the filter parameters can be tuned to match the lines, which are in different position for each class.

Algorithm (6/6)

Feature Extraction



- A global **feature** is extracted from the output image of each filter by **adding up the values of all pixels**

$$F_q = g_{l,m} = \sum_{\nu}\sum_{\varepsilon} |g(\nu,\varepsilon; f_l,\theta_m)|$$

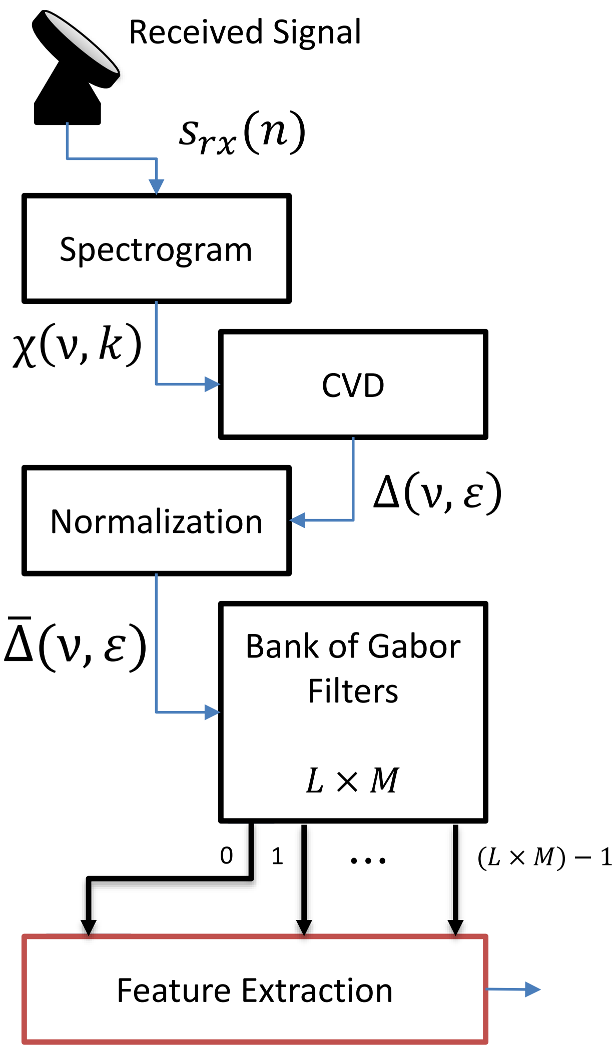
Where $\nu = 0,\cdots,N_1-1, \quad \varepsilon = 0,\cdots,N_2-1,$

$q = mL + l,$

with $l = 0,\cdots,L-1, \quad \mathrm{m} = 0,\cdots,M-1,$

Algorithm (6/6)

Feature Extraction



- A global **feature** is extracted from the output image of each filter by **adding up the values of all pixels**

$$F_q = g_{l,m} = \sum_{\nu} \sum_{\varepsilon} |g(\nu, \varepsilon; f_l, \theta_m)|$$

Where $\nu = 0, \cdots, N_1 - 1, \quad \varepsilon = 0, \cdots, N_2 - 1,$

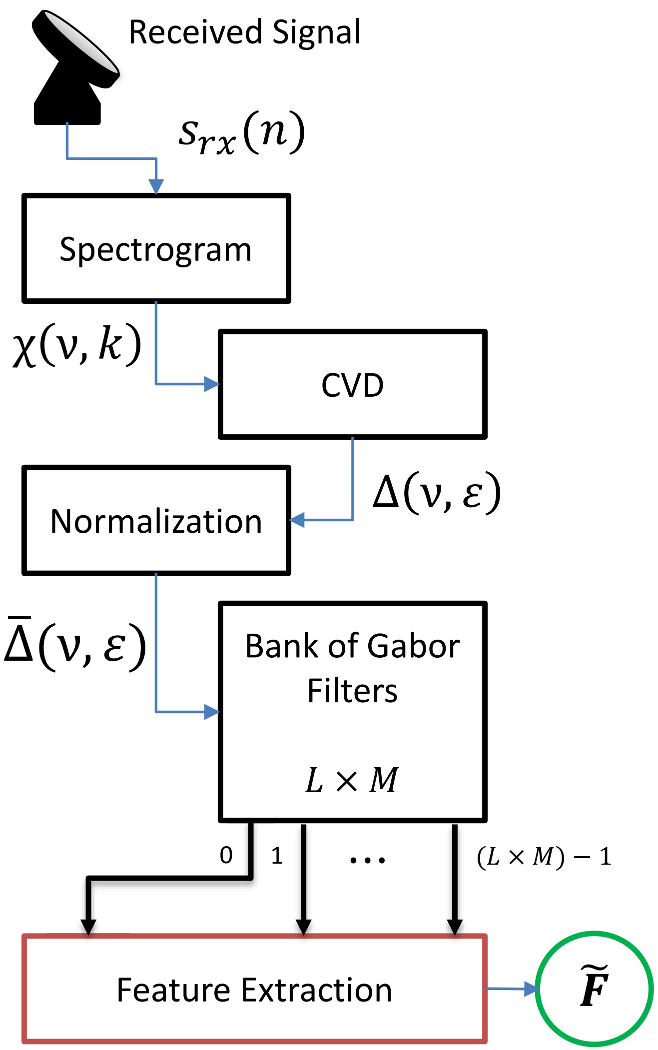
$q = mL + l,$

with $l = 0, \cdots, L - 1, \quad \mathrm{m} = 0, \cdots, M - 1,$

- The **Feature Vector** is $\boldsymbol{F} = \left[F_0 \;\; F_1 \;\; \cdots \;\; F_{(L \times M) - 1}\right]$

Algorithm (6/6)

Feature Extraction



- A global **feature** is extracted from the output image of each filter by **adding up the values of all pixels**

$$F_q = g_{l,m} = \sum_{\nu} \sum_{\varepsilon} |g(\nu, \varepsilon; f_l, \theta_m)|$$

Where $\nu = 0, \cdots, N_1 - 1, \quad \varepsilon = 0, \cdots, N_2 - 1,$

$q = mL + l,$

with $l = 0, \cdots, L - 1, \quad \mathrm{m} = 0, \cdots, M - 1,$

- The **Feature Vector** is $\boldsymbol{F} = \left[F_0 \;\; F_1 \;\; \cdots \;\; F_{(L \times M)-1}\right]$

- The **Feature Vector** is normalised before it is used in the classifier as follows:

$$\widetilde{\boldsymbol{F}} = \frac{\boldsymbol{F} - \eta_{\boldsymbol{F}}}{\sigma_{\boldsymbol{F}}}$$

where $\eta_{\boldsymbol{F}}$ and $\sigma_{\boldsymbol{F}}$ are the statistical mean and standard deviation of the vector F, respectively.

Performance (1/3)

Real Data

- The algorithm has been tested on **real data**;

Performance (1/3)

Real Data

- The algorithm has been tested on **real data**;
- The data set has been realized acquiring the signals scattered by the **replicas** of the targets by a **Continuous Wave (CW) radar**, on varying the angles of azimuth and elevation;

Performance (1/3)

Real Data

- The algorithm has been tested on **real data**;
- The data set has been realized acquiring the signals scattered by the **replicas** of the targets by a **Continuous Wave (CW) radar**, on varying the angles of azimuth and elevation;
- The different movements of warheads and decoys have been simulated by **using ST robotic manipulator R-17** and an added rotor.

Performance (1/3)

Real Data

- The algorithm has been tested on **real data**;
- The data set has been realized acquiring the signals scattered by the **replicas** of the targets by a **Continuous Wave (CW) radar**, on varying the angles of azimuth and elevation;
- The different movements of warheads and decoys have been simulated by **using ST robotic manipulator R-17** and an added rotor.

replicas

PERFORMANCE (1/3)

REAL DATA

- The algorithm has been tested on **real data**;
- The data set has been realized acquiring the signals scattered by the **replicas** of the targets by a **Continuous Wave (CW) radar**, on varying the angles of azimuth and elevation;
- The different movements of warheads and decoys have been simulated by **using ST robotic manipulator R-17** and an added rotor.

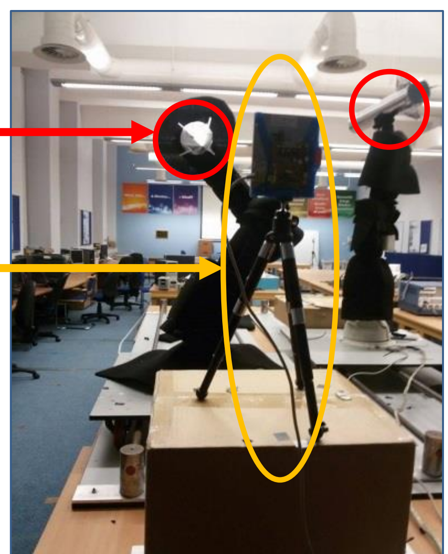
PERFORMANCE (1/3)

REAL DATA

- The algorithm has been tested on **real data**;
- The data set has been realized acquiring the signals scattered by the **replicas** of the targets by a **Continuous Wave (CW) radar**, on varying the angles of azimuth and elevation;
- The different movements of warheads and decoys have been simulated by **using ST robotic manipulator R-17** and an added rotor.

replicas

radar



robotic manipulator

PERFORMANCE (2/3)

CLASSIFIER AND FIGURES OF MERITS

- The performance is evaluated using ***k*-Nearest Neighbour** (kNN) classifier.

Performance (2/3)

Classifier and Figures of Merits

- The performance is evaluated using **k-Nearest Neighbour** (kNN) classifier.
- **3** figures of merits are considered.

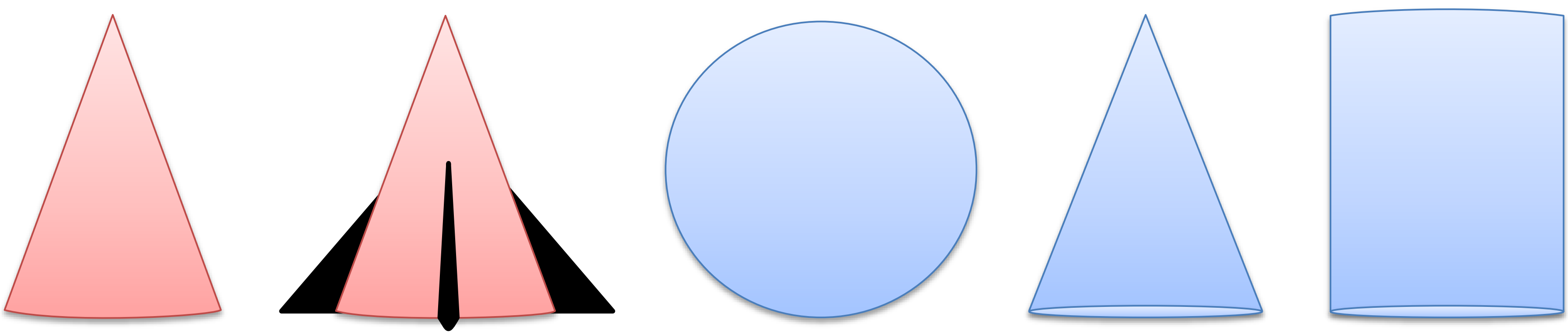
PERFORMANCE (2/3)

CLASSIFIER AND FIGURES OF MERITS

- The performance is evaluated using ***k*-Nearest Neighbour** (kNN) classifier.
- **3** figures of merits are considered.
- **Probability of correct Classification** (P_C), which represents the capability to distinguish among the **warhead class** and the **decoy class**.

warhead class

decoy class



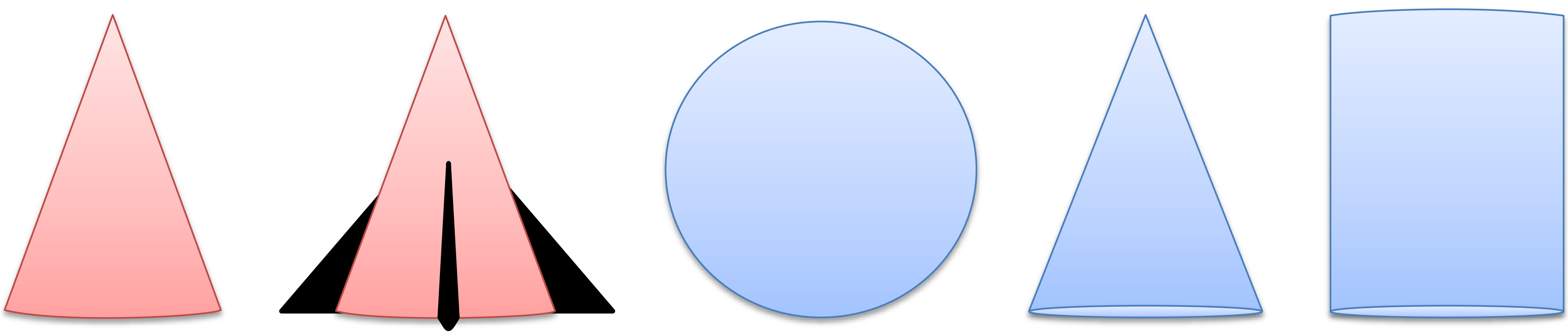
PERFORMANCE (2/3)

CLASSIFIER AND FIGURES OF MERITS

- The performance is evaluated using ***k*-Nearest Neighbour** (kNN) classifier.
- **3** figures of merits are considered.

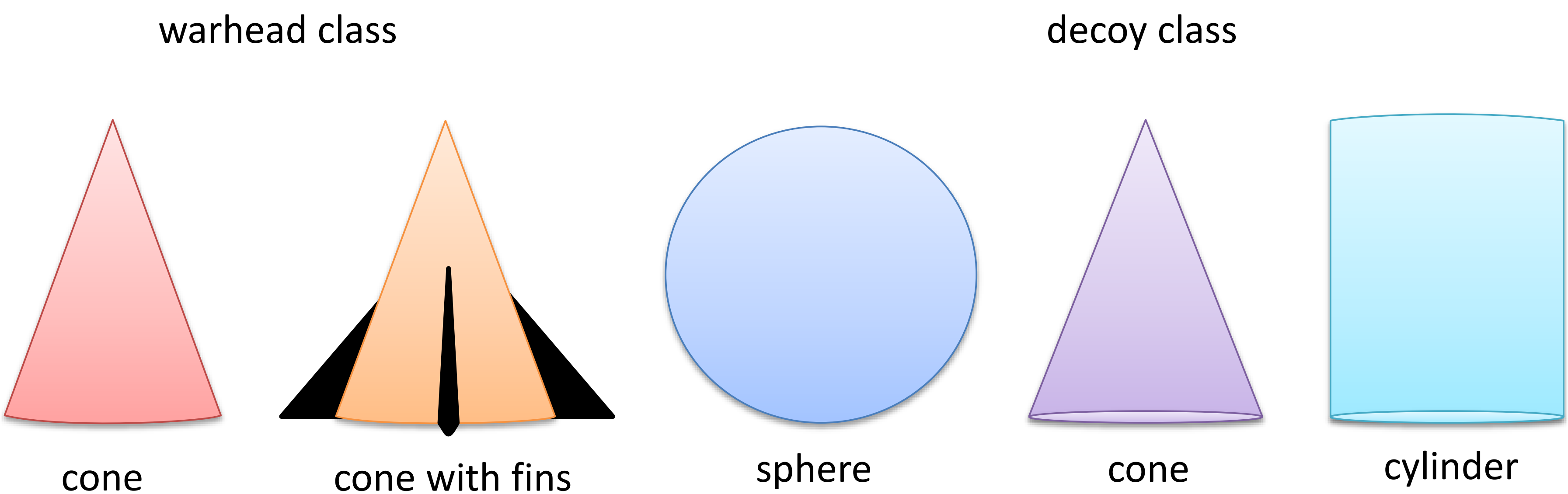
warhead class

decoy class



Performance (2/3)
Classifier and Figures of Merits

- The performance is evaluated using ***k*-Nearest Neighbour** (kNN) classifier.
- **3** figures of merits are considered.
- **Probability of correct Recognition** (P_R), which represents the capability to identify the actual **shape** of the target within the warhead and the decoy class.



PERFORMANCE (2/3)

CLASSIFIER AND FIGURES OF MERITS

- The performance is evaluated using **k-Nearest Neighbour** (kNN) classifier.
- **3** figures of merits are considered.

Performance (2/3)

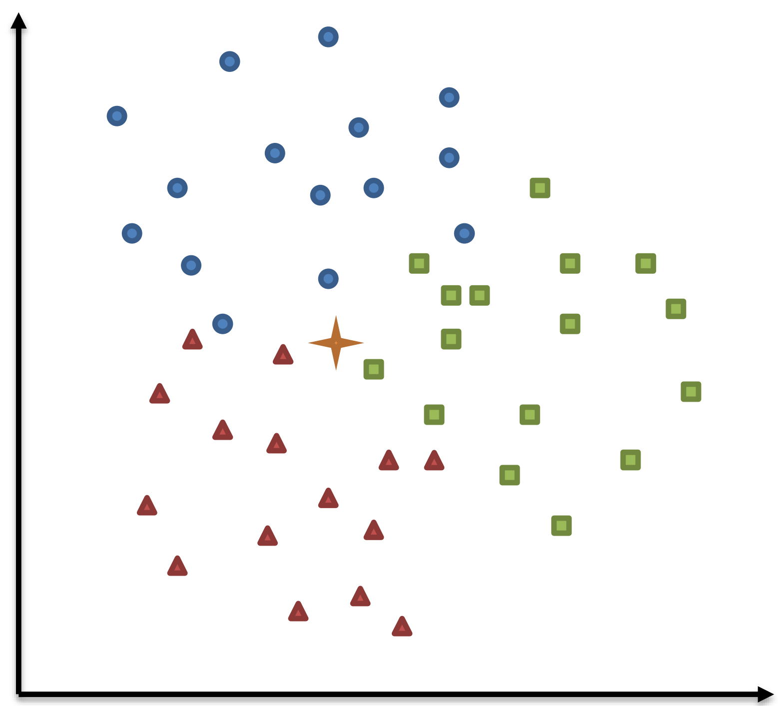
Classifier and Figures of Merits

- The performance is evaluated using **k-Nearest Neighbour** (kNN) classifier.
- **3** figures of merits are considered.
- **Probability of Unknown** (P_U), which is given by the ratio of the number of analysed objects for which the classifier **does not take a decision** and their total number.

Performance (2/3)

Classifier and Figures of Merits

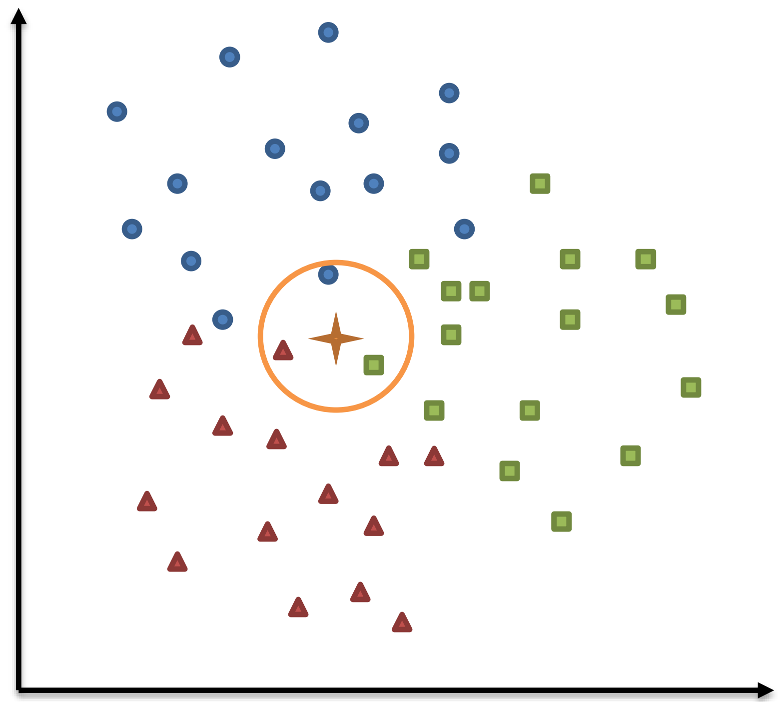
- The performance is evaluated using **<u>k-Nearest Neighbour</u>** (<u>k</u>NN) classifier.
- **<u>3</u>** <u>figures of merits</u> are considered.
- **Probability of Unknown** (P_U), which is given by the ratio of the number of analysed objects for which the classifier **<u>does not take a decision</u>** and their total number.
    - Let us consider the shown example with 3 classes and $k = 3$.



Performance (2/3)

Classifier and Figures of Merits

- The performance is evaluated using ***k*-Nearest Neighbour** (kNN) classifier.
- **3** figures of merits are considered.
- **Probability of Unknown** (P_U), which is given by the ratio of the number of analysed objects for which the classifier **does not take a decision** and their total number.
  - Let us consider the shown example with 3 classes and $k = 3$.
  - Each of the 3 nearest neighbour belong to different classes.



PERFORMANCE (2/3)

CLASSIFIER AND FIGURES OF MERITS

- The performance is evaluated using **k-Nearest Neighbour** (kNN) classifier.

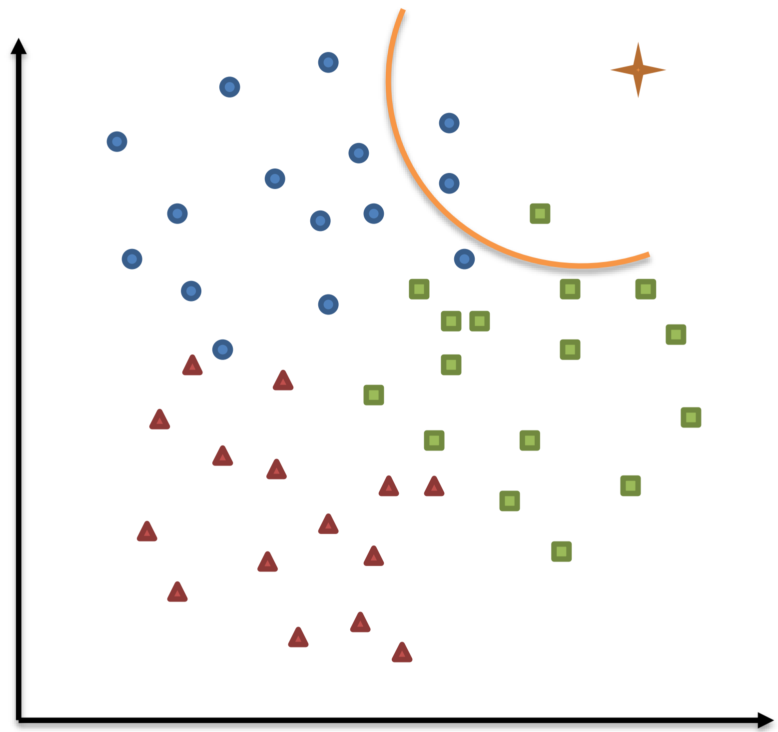
- **3** figures of merits are considered.

- **Probability of Unknown** (P_U), which is given by the ratio of the number of analysed objects for which the classifier **does not take a decision** and their total number.

  - Let us consider the shown example with 3 classes and $k = 3$.

  - Each of the 3 nearest neighbour belong to different classes.

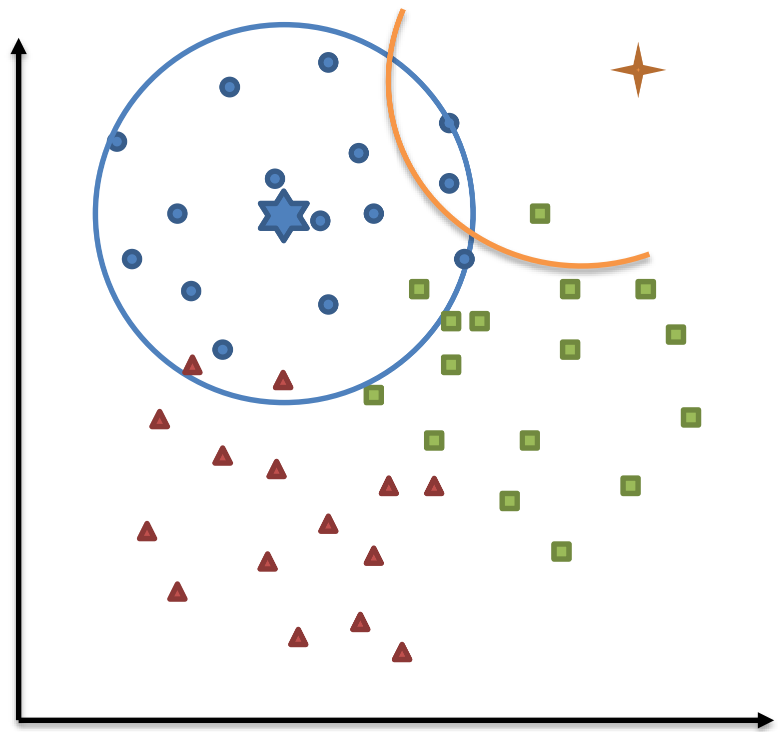
  - The tested vector is too far from the hyperspheres made by the training vectors of each class.



Performance (2/3)

Classifier and Figures of Merits

- The performance is evaluated using **<u>k-Nearest Neighbour</u>** (kNN) classifier.
- **<u>3</u>** <u>figures of merits</u> are considered.
- **Probability of Unknown** (P_U), which is given by the ratio of the number of analysed objects for which the classifier **<u>does not take a decision</u>** and their total number.
  - Let us consider the shown example with 3 classes and $k = 3$.
  - Each of the 3 nearest neighbour belong to different classes.
  - The tested vector is too far from the **<u>hyperspheres</u>** made by the training vectors of each class.



PERFORMANCE (3/3)

RESULTS

- A Monte Carlo approach has been used over 50 tests:
  - The **mean** of the three figures of merit is evaluated.
  - the available samples have been divided randomly with 70% used for training and the other 30% for testing.

PERFORMANCE (3/3)

RESULTS

- A Monte Carlo approach has been used over 50 tests:
  - The **mean** of the three figures of merit is evaluated.
  - the available samples have been divided randomly with 70% used for training and the other 30% for testing.

- The algorithm is tested with respect to the variation of:

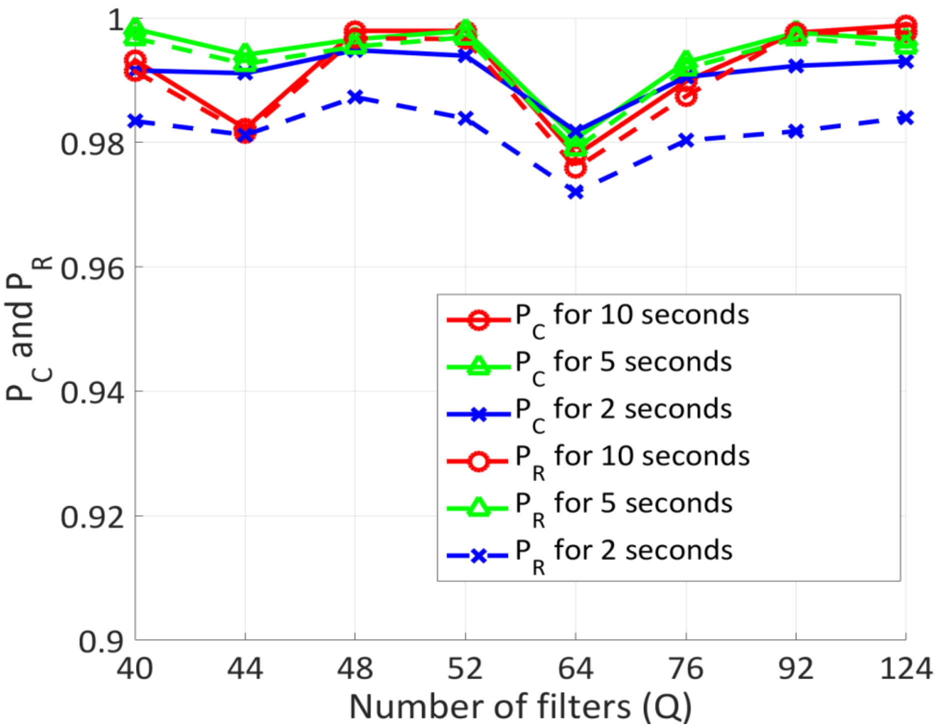
  - the available **observation time**;

  - the **dimension** of bank of filters;

Performance (3/3)

Results

- A Monte Carlo approach has been used over 50 tests:
  - The **mean** of the three figures of merit is evaluated.
  - the available samples have been divided randomly with 70% used for training and the other 30% for testing.
- The algorithm is tested with respect to the variation of:
  - the available **observation time**;
  - the **dimension** of bank of filters;



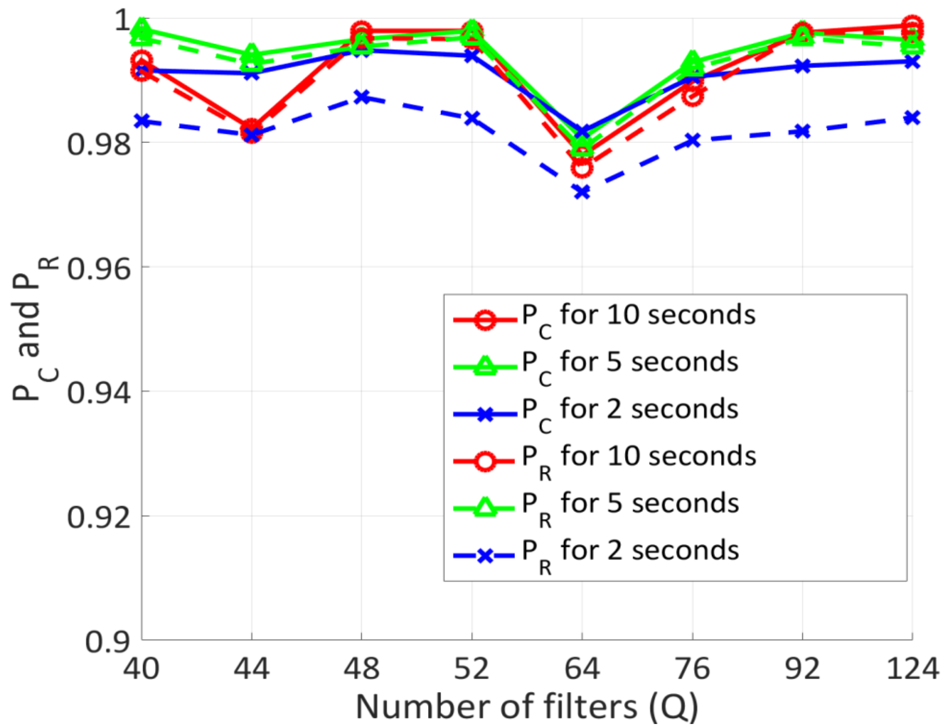
PERFORMANCE (3/3)

RESULTS

- A Monte Carlo approach has been used over 50 tests:
  - The **mean** of the three figures of merit is evaluated.
  - the available samples have been divided randomly with 70% used for <u>training</u> and the other 30% for <u>testing</u>.

- The algorithm is tested with respect to the variation of:
  - <u>the available **observation time**;</u>
  - <u>the **dimension** of bank of filters;</u>

P_U is smaller than 0.02



Performance (3/3)

Results

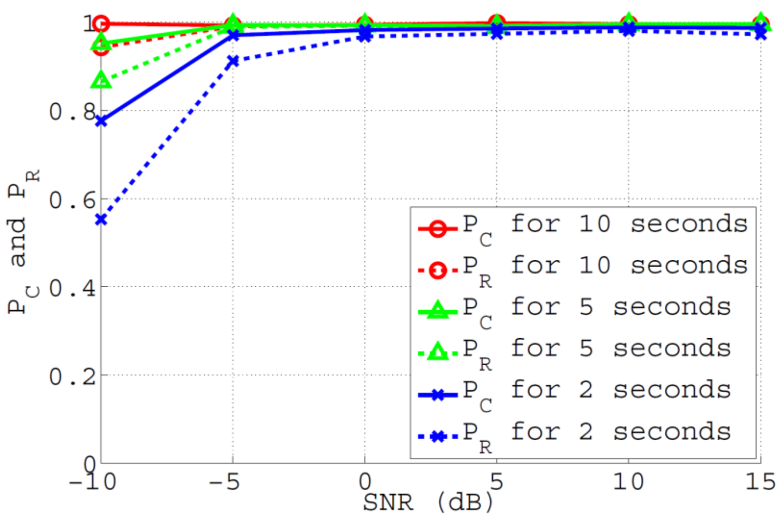
- Fixing the number of filters equal to 124, the algorithm is tested with respect to the variation of observation time and SNR;

PERFORMANCE (3/3)

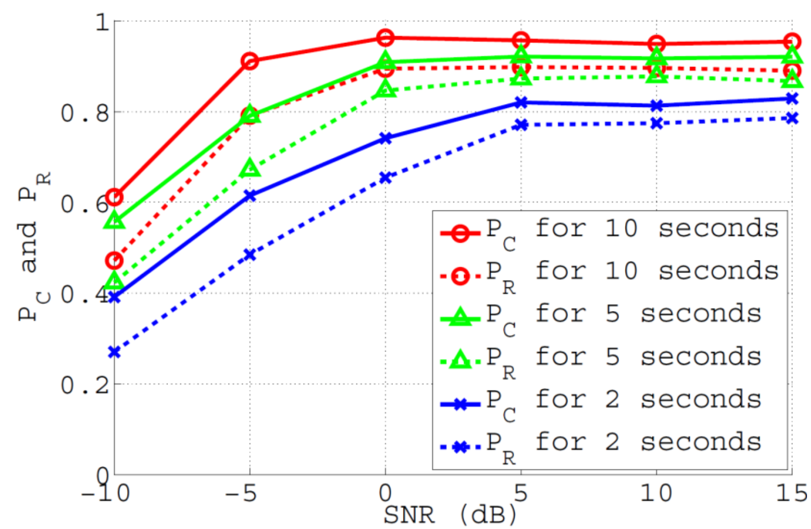
RESULTS

- Fixing the number of filters equal to 124, the algorithm is tested with respect to the variation of observation time and SNR;
- It has been compared with our pseudo-Zernike moments based approach[1], with moments order equal to 10 (<u>feature vector dimension = 121</u>);

Performance for Gabor filter based approach



Performance for pZ moments based approach



[1] C. Clemente, L. Pallotta, A. De Maio, J. Soraghan, and A. Farina, "A novel algorithm for radar classification based on Doppler characteristics exploiting orthogonal pseudo-Zernike polynomials," IEEE Transactions on Aerospace and Electronic Systems, vol. 51, no. 1, pp. 417–430, 2015.

Performance (3/3)

Results

- Fixing the number of filters equal to 124, the algorithm is tested with respect to the variation of observation time and SNR;
- It has been compared with our pseudo-Zernike moments based approach[1], with moments order equal to 10 (<u>feature vector dimension = 121</u>);

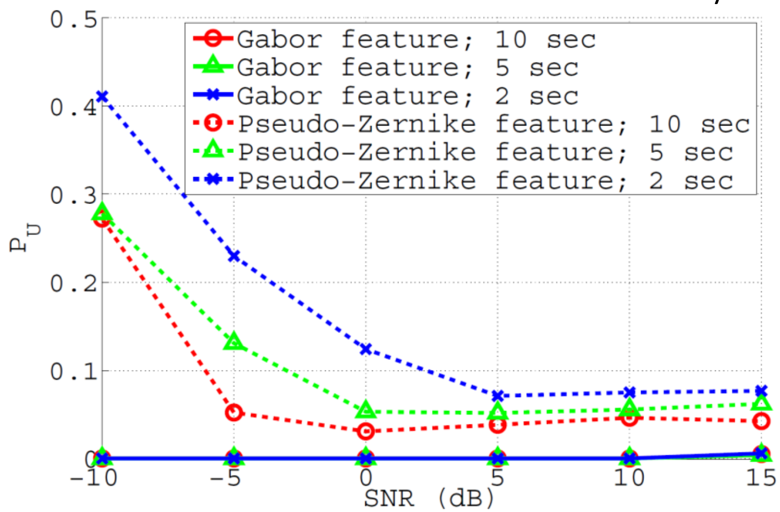
[1] C. Clemente, L. Pallotta, A. De Maio, J. Soraghan, and A. Farina, "A novel algorithm for radar classification based on Doppler characteristics exploiting orthogonal pseudo-Zernike polynomials," IEEE Transactions on Aerospace and Electronic Systems, vol. 51, no. 1, pp. 417–430, 2015.

PERFORMANCE (3/3)

RESULTS

- Fixing the number of filters equal to 124, the algorithm is tested with respect to the variation of observation time and SNR;
- It has been compared with our pseudo-Zernike moments based approach[1], with moments order equal to 10 (<u>feature vector dimension = 121</u>);

Performance in terms of Unknown Probability



[1] C. Clemente, L. Pallotta, A. De Maio, J. Soraghan, and A. Farina, "A novel algorithm for radar classification based on Doppler characteristics exploiting orthogonal pseudo-Zernike polynomials," IEEE Transactions on Aerospace and Electronic Systems, vol. 51, no. 1, pp. 417–430, 2015.

Conclusions and Future Plans

- A novel classification algorithm that is able to differentiate between targets of interest and interference factors was presented.
- The algorithm is based on the using of 2-D Gabor Filter.
- The algorithm takes advantage from the FFT algorithm.
- The features are robust with respect to the noise.
- The performance is satisfactory also for low feature vector dimension.
- The approach was tested on real data with success.

Thank you!

Any Question?

University of
Strathclyde
Glasgow