

Fourier transform, in 1D and in 2D

Václav Hlaváč

Czech Technical University in Prague

Czech Institute of Informatics, Robotics and Cybernetics

166 36 Prague 6, Jugoslávských partyzánů 1580/3, Czech Republic

<http://people.ciirc.cvut.cz/hlavac>, vaclav.hlavac@cvut.cz

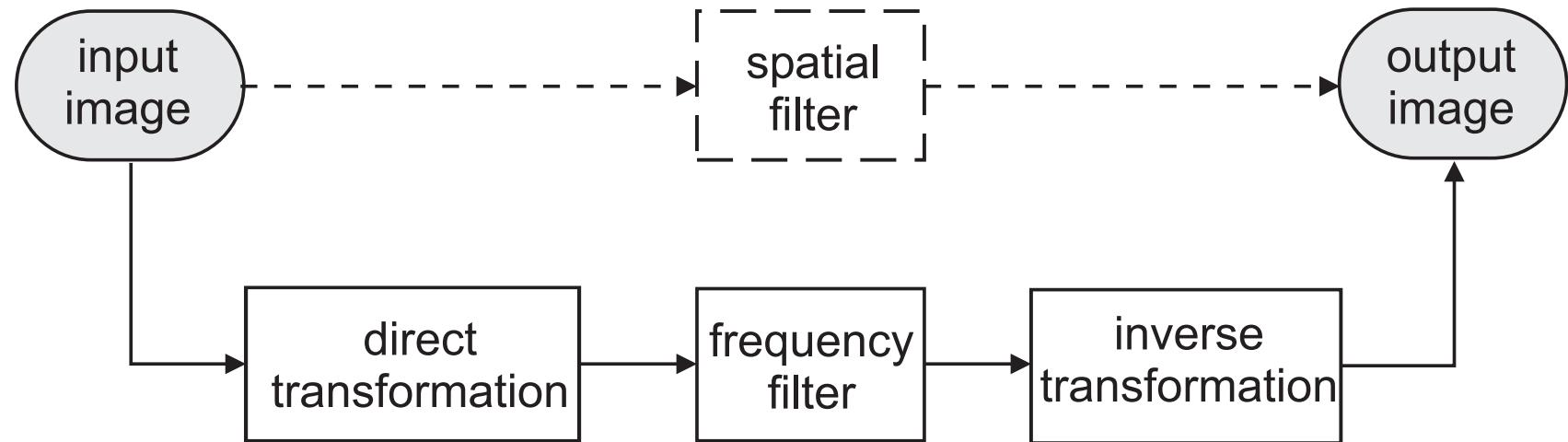
also Center for Machine Perception, <http://cmp.felk.cvut.cz>

Outline of the talk:

- ◆ Fourier tx in 1D, computational complexity, FFT.
- ◆ Fourier tx in 2D, centering of the spectrum.
- ◆ Examples in 2D.

Initial idea, filtering in frequency domain

Image processing \equiv filtration of 2D signals.



Filtration in the spatial domain. We would say in time domain for 1D signals. It is a linear combination of the input image with coefficients of (often local) filter. The basic operation is called convolution.

Filtration in the frequency domain. Conversion to the ‘frequency domain’, filtration there, and the conversion back.

We consider *Fourier transform*, but there are other linear integral transforms serving a similar purpose, e.g., cosine, wavelets.

1D Fourier transform, introduction

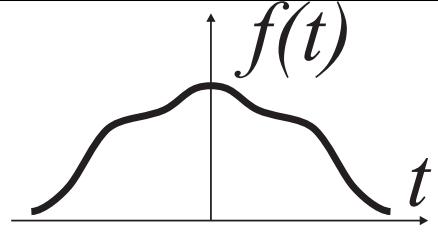
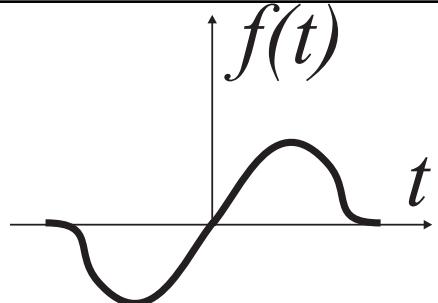
- ◆ Fourier transform is one of the most commonly used techniques in (linear) signal processing and control theory.
- ◆ It provides one-to-one transform of signals from/to a time-domain representation $f(t)$ to/from a frequency domain representation $F(\xi)$.
- ◆ It allows a frequency content (spectral) analysis of a signal.
- ◆ FT is suitable for periodic signals.
- ◆ If the signal is not periodic then the Windowed FT or the linear integral transformation with time (spatially in 2D) localized basis function, e.g., wavelets, Gabor filters can be used.



Joseph Fourier

1768-1830

Odd, even and complex conjugate functions

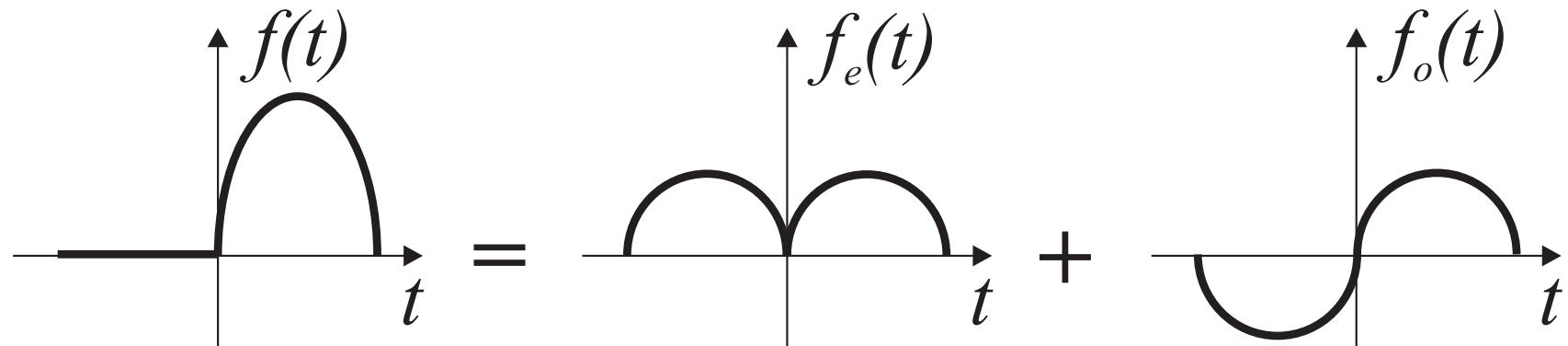
Even	$f(t) = f(-t)$	
Odd	$f(t) = -f(-t)$	
Conjugate symmetric	$f(\xi) = f^*(-\xi)$	$f(-5) = 2 + 3i$ $f(-5) = 2 - 3i$

- ◆ f^* denotes a complex conjugate function.
- ◆ i is a complex unit.

Any function can be decomposed
as a sum of the even and odd part

$$f(t) = f_e(t) + f_o(t)$$

$$f_e(t) = \frac{f(t) + f(-t)}{2} \quad f_o(t) = \frac{f(t) - f(-t)}{2}$$



Fourier Tx definition: continuous cased

$\mathcal{F}\{f(t)\} = F(\xi)$, where ξ [Hz= s^{-1}] is a frequency and $2\pi\xi$ [s^{-1}] is the angular frequency.

Fourier Tx	Inverse Fourier Tx
$F(\xi) = \int_{-\infty}^{\infty} f(t) e^{-2\pi i \xi t} dt$	$f(t) = \int_{-\infty}^{\infty} F(\xi) e^{2\pi i \xi t} d\xi$

What is the meaning of the inverse Fourier Tx? Express it as a Riemann sum:

$$f(t) \doteq (\dots + F(\xi_0) e^{2\pi i \xi_0 t} + F(\xi_1) e^{2\pi i \xi_1 t} + \dots) \Delta\xi ,$$

kde $\Delta\xi = \xi_{k+1} - \xi_k$ pro $\forall k$.

⇒ Any 1D function can be expressed as a the weighted sum (integral) of many different complex exponentials (because of Euler's formula $e^{i\xi} = \cos \xi + i \sin \xi$, also of cosinusoids and sinusoids).

Existence conditions of Fourier Tx

1. $\int_{-\infty}^{\infty} |f(t)| dt < \infty$, i.e. $f(t)$ has to grow slower than an exponential curve.
2. $f(t)$ can have only a finite number of discontinuities and maxima, minima in any finite rectangle.
3. $f(t)$ need not have discontinuities with the infinite amplitude.

Fourier transformation exists always for digital images as they are limited and have finite number of discontinuities.

Fourier Tx, symmetries

- ◆ Symmetry with regards to the complex conjugate part, i.e.,
 $F(-i\xi) = F^*(i\xi)$.
- ◆ $|F(i\xi)|$ is always even.
- ◆ The phase of $F(i\xi)$ is always odd.
- ◆ $Re\{F(i\xi)\}$ is always even.
- ◆ $Im\{F(i\xi)\}$ is always odd.
- ◆ The even part of $f(t)$ transforms to the real part of $F(i\xi)$.
- ◆ The odd part of $f(t)$ transforms to the imaginary part of $F(i\xi)$.

Convolution, definition, continuous case

- ◆ Convolution (in functional analysis) is an operation on two functions f and h , which produces a third function ($f * h$), often used to create a modification of one of the input functions.
- ◆ Convolution is an integral ‘mixing’ values of two functions, i.e., of the function $h(t)$, which is shifted and overlayed with the function $f(t)$ or vice-versa.
- ◆ Consider first the **continuous case** with general infinite limits

$$(f * h)(t) = (h * f)(t) \equiv \int_{-\infty}^{\infty} f(\tau) h(t - \tau) d\tau = \int_{-\infty}^{\infty} f(t - \tau) h(\tau) d\tau .$$

- ◆ The limits can be constraint to the interval $[0, t]$, because we assume zero values of functions for the negative argument

$$(f * h)(t) = (h * f)(t) \equiv \int_0^t f(\tau) h(t - \tau) d\tau = \int_0^t f(t - \tau) h(\tau) d\tau .$$

Cross-correlation and convolution

Convolution $*$ defined for 1D signals uses the flipped kernel h

$$(f * h)(t) \equiv \int_{-\infty}^{\infty} f(\tau) h(t - \tau) d\tau .$$

Cross-correlation \star defined for 1D signals uses the (unflipped) kernel h

$$(f \star h) \equiv \int_{-\infty}^{\infty} f^*(\tau) h(t + \tau) d\tau ,$$

where f^* denotes the complex conjugate of f .

Cross-correlation is a measure of similarity of two functions as a function of a (time) shift τ .

Convolution, discrete approximation

$$(f * h)(i) = (h * f)(i) \equiv \sum_{m \in \mathcal{O}} h(i - m) f(m) = \sum_{m \in \mathcal{O}} h(i) f(i - m),$$

where \mathcal{O} is a local neighborhood of a ‘current position’ i and h is the convolution kernel (also convolution mask).

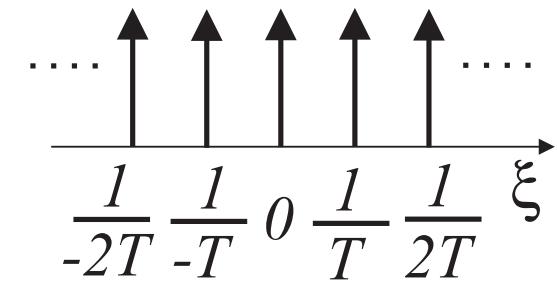
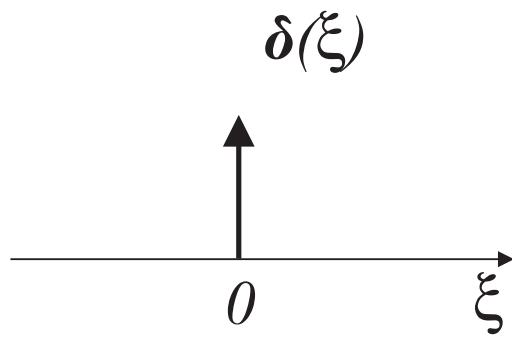
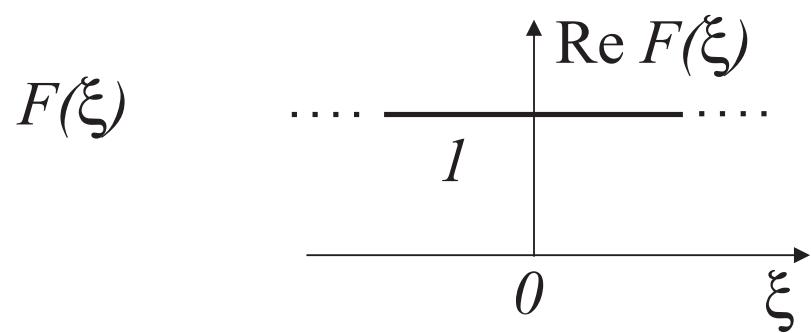
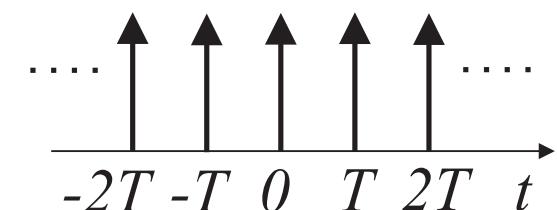
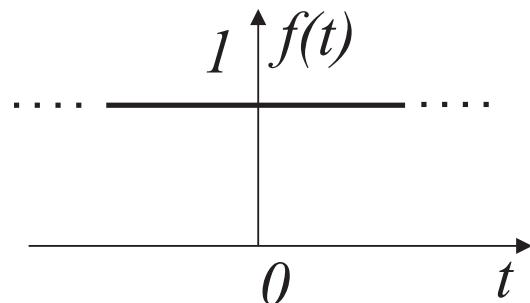
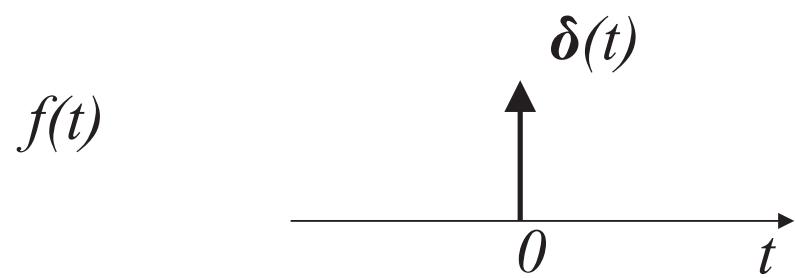
Fourier Tx, properties (1)

Property	$f(t)$	$F(\xi)$
Linearity	$a f_1(t) + b f_2(t)$	$a F_1(\xi) + b F_2(\xi)$
Duality	$F(t)$	$f(-\xi)$
Convolution	$(f * g)(t)$	$F(\xi) G(\xi)$
Product	$f(t) g(t)$	$(F * G)(\xi)$
Time shift	$f(t - t_0)$	$e^{-2\pi i \xi t_0} F(\xi)$
Frequency shift	$e^{2\pi i \xi_0 t} f(t)$	$F(\xi - \xi_0)$
Differentiation	$\frac{df(t)}{dt}$	$2\pi i \xi F(\xi)$
Multiplication by t	$t f(t)$	$\frac{i}{2\pi} \frac{dF(\xi)}{d\xi}$
Time scaling	$f(a t)$	$\frac{1}{ a } F(\xi/a)$

Fourier Tx, properties (2)

Area in time	$F(0) = \int_{-\infty}^{\infty} f(t)dt$	Area function $f(t)$.
Area in freq.	$f(0) = \int_{-\infty}^{\infty} F(\xi)d\xi$	Area under $F(\xi)$
Parseval's th.	$\int_{-\infty}^{\infty} f(t) ^2 dt = \int_{-\infty}^{\infty} F(\xi) ^2 d\xi$	f energy = F energy

Basic Fourier Tx pairs (1)

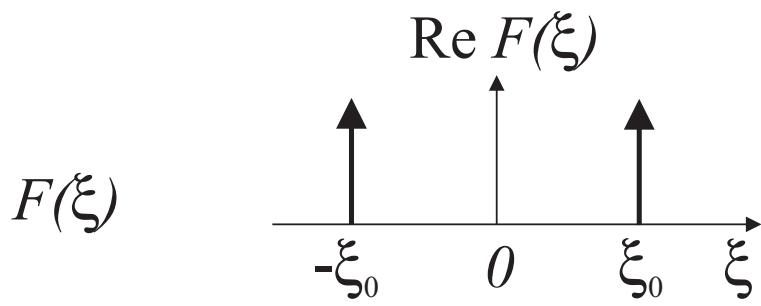
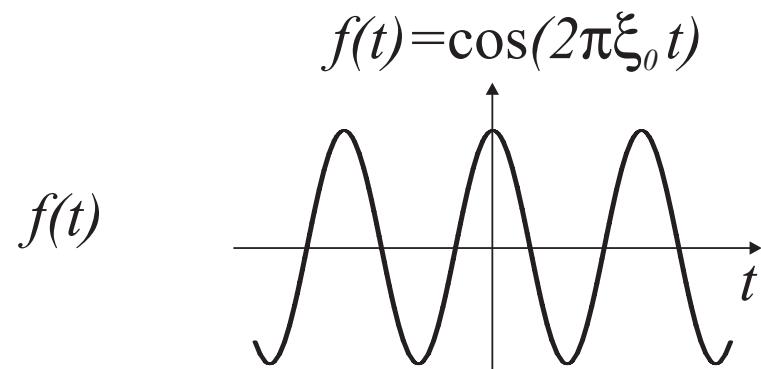


Dirac

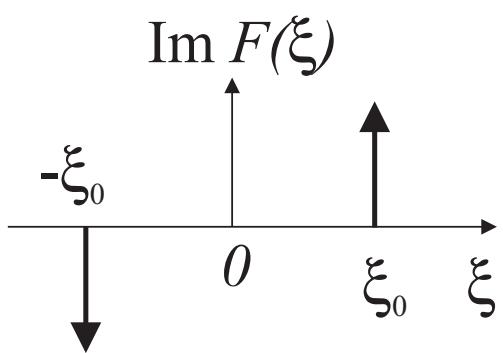
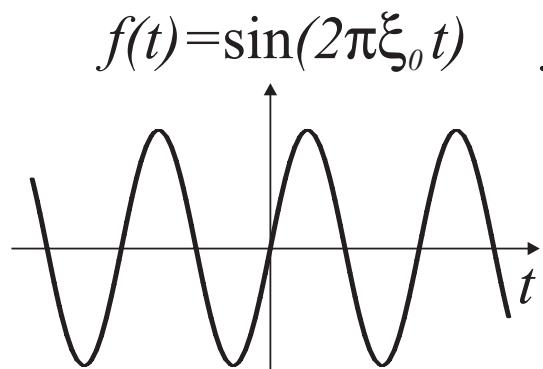
constant

∞ sequence of Diracs

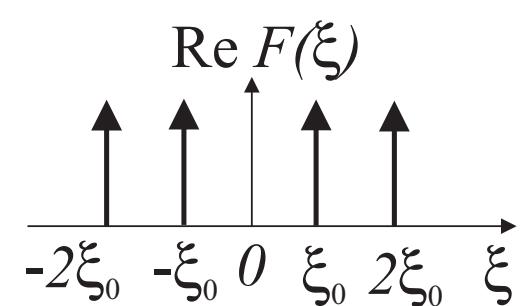
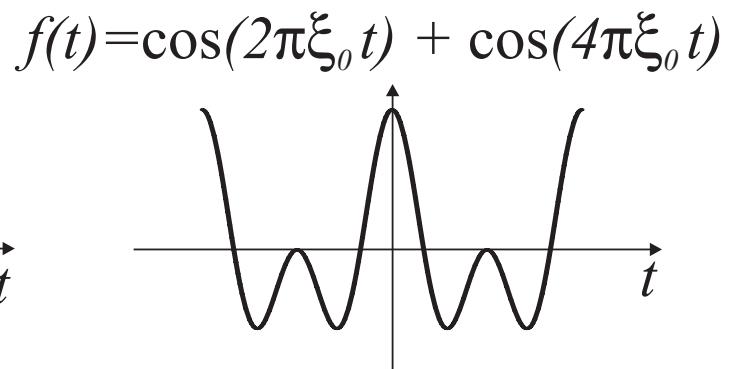
Basic Fourier Tx pairs (2)



cosine

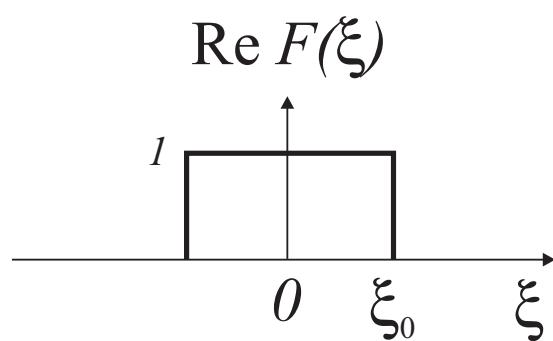
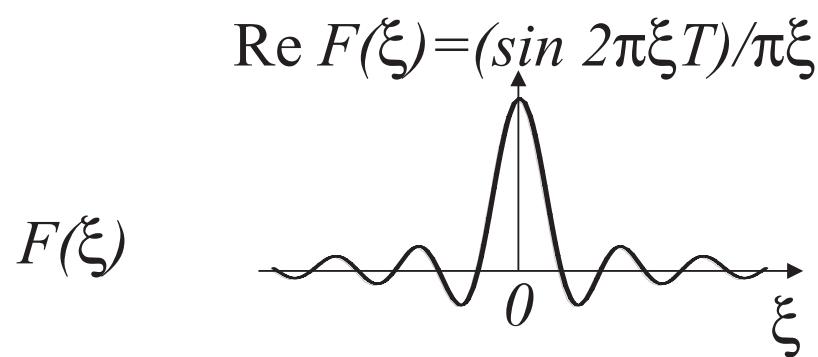
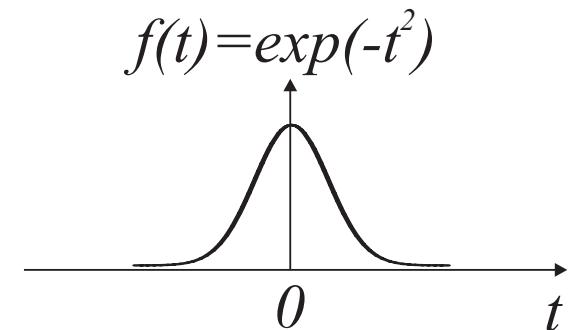
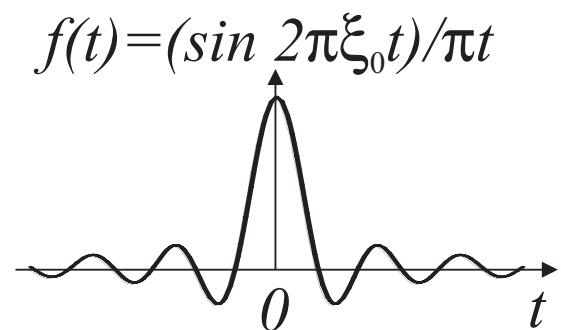
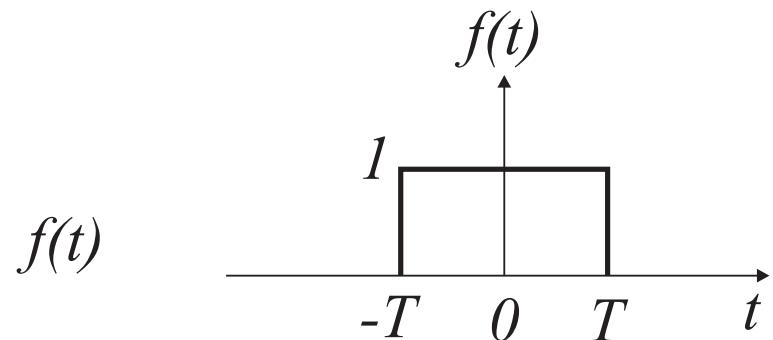


sine



two cosines mixture

Basic Fourier Tx pairs (3)



rectangle in t

rectangle in ξ

Gaussian

Uncertainty principle

- ◆ All Fourier Tx pairs are constrained by the uncertainty principle.
- ◆ The signal of short duration must have wide Fourier spectrum and vice versa.
- ◆ $(\text{signal duration}) (\text{frequency bandwith}) \geq \frac{1}{\pi}$
- ◆ Observation: Gaussian e^{-t^2} modulated by a sinusoid (Gabor function) has the smallest duration-bandwidth product.
- ◆ The principle is an instance of the general uncertainty principle introduced by Werner Heisenberg in quantum mechanics.

Non-periodic signals

Fourier transform assumes a periodic signal. What if a non-periodic signal has to be processed? There are two common approaches.

1. To process the signal in small chunks (windows) and assume that the signal is periodic outside the windows.

- ◆ The approach was introduced by Dennis Gabor in 1946 and it is named Short time Fourier transform.

Dennis Gabor, 1900-1979, inventor of holography, Nobel price for physics in 1971, studied in Budapest, PhD in Berlin in 1927, fled Nazi persecution to Britain in 1933.

- ◆ Mere cutting of the signal to rectangular windows is not good because discontinuities at windows limits cause unwanted high frequencies.
- ◆ This is the reason why the signal is convolved by a damping weight function, often Gaussian or Hamming function ensuring the zero signal value at the limits of the window and beyond it.

2. Use of more complex basis function, e.g., wavelets in the wavelet transform.

Discrete Fourier transform

- ◆ Let $f(n)$ be an input signal (a sequence), $n = 0, \dots, N - 1$.
- ◆ Let $F(k)$ be a Frequency spectrum (the result of the discrete Fourier transformation) of a signal $f(n)$.
- ◆ Discrete Fourier transformation

$$F(k) \equiv \sum_{n=0}^{N-1} f(n) e^{-\frac{2\pi i k n}{N}}$$

- ◆ Inverse discrete Fourier transformation

$$f(n) \equiv \frac{1}{N} \sum_{k=0}^{N-1} F(k) e^{\frac{2\pi i k n}{N}}$$

Computational complexity, a reminder

- ◆ While considering complexity, it is abstracted from a specific computer. Only an asymptotic behavior of algorithms is concerned.
- ◆ Bounds are sought, which are used to express time or memory requirements of an algorithm.
- ◆ An asymptotic upper and lower bounds for the magnitude of a function $g(n)$ (i.e., its growth) in terms of another, usually simpler, function is sought.
- ◆ $\mathcal{O}(g(n))$ denotes the set of functions $f(n)$, where $f(n)$ bounds $g(n)$ asymptotically from below. Formally, there exist a positive constant c and a number n_0 such that $0 \leq f(n) \leq c g(n)$ for all $n \geq n_0$.
- ◆ $\Omega(g(n))$ denotes the set of functions $f(n)$, where bounds $g(n)$ asymptotically from above. Formally, there exist a positive constant c and a number n_0 such that $0 \leq c g(n) \leq f(n)$ for all $n \geq n_0$.

Computational complexity, the notation

- ◆ ‘Big $\mathcal{O}()$ ’ notation; for example, $\mathcal{O}(n^2)$ means that the number of algorithm steps will be roughly proportional to the square of the number of samples in the worst case.
- ◆ Additional terms and multiplicative constants are not taken into account because a qualitative comparison is sought.
- ◆ The quadratic complexity $\mathcal{O}(n^2)$ is worse than say $\mathcal{O}(n)$ (linear) or $\mathcal{O}(1)$ (constant, independent of the length n), but is better than $\mathcal{O}(n^3)$ (cubic).
- ◆ If the complexity is exponential, e.g., $\mathcal{O}(2^n)$, then it often means that the algorithm cannot be applied to larger problems (in practical terms).
- ◆ Similarly, $\Omega()$ notation.

Computational complexity of the discrete Fourier transform

- ◆ Let W be a complex number, $W \equiv e^{\frac{-2\pi i}{N}}$.

$$F(k) \equiv \sum_{n=0}^{N-1} f(n) e^{\frac{-2\pi i kn}{N}} = \sum_{n=0}^{N-1} W^{nk} f(n)$$

- ◆ The vector $f(n)$ is multiplied by the matrix whose element (n, k) is the complex constant W to the power $N \cdot k$.
- ◆ This has the computational complexity $\mathcal{O}(N^2)$.

Fast Fourier transform

- ◆ A fast Fourier transform (FFT) is an efficient algorithm to compute the discrete Fourier transform and its inverse.
- ◆ Statement: FFT has the complexity $\mathcal{O}(N \log_2 N)$.
- ◆ [Example](#) (according to Numerical recepies in C):
 - A sequence of $N = 10^6$, 1 μ second computer.
 - FFT 30 seconds of CPU time.
 - DFT 2 weeks of CPU time, i.e., 1,209,600 seconds, which is about $40.000 \times$ more.

Fast Fourier transform, the signal division

◆ A FFT core idea

The DFT of length N can be expressed as sum of two DFTs of length $N/2$, the first one consisting of **odd** and the second of **even** samples.

(Danielson, Lanczos in 1942; further developed by Cooley, Tukey in 1965)

◆ There are two approaches how to split the signal called

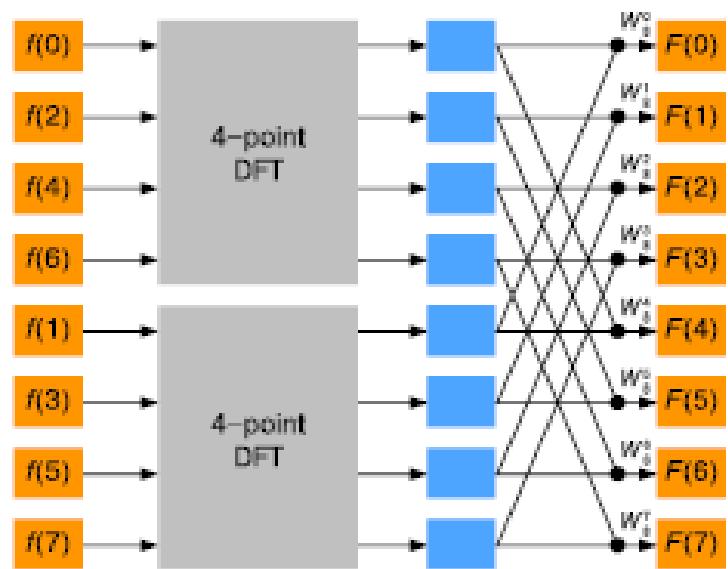
- Decimation in time (DIT);
- Decimation in frequency (DIF).

◆ Note 1: *FFT exists also for a general length N .*

◆ Note 2: *The input sequence can be divided to more than two parts in general.*

Decimation in time (DIT)

- ◆ The input sequence $f(n)$, $n = 1, \dots, N - 1$ is divided into even $f^e(n')$ and odd $f^o(n')$ parts, $n' = 0, 1, \dots, N/2 - 1$.
- ◆ The Fourier transform of corresponding parts denoted F^e , F^o can be calculated recursively $F(k) = F^e(k) + W^{Nk} F^o(k)$, where $k = 0, 1, \dots, N$.
- ◆ The signals F^e and F^o are of a half length. Due to their periodicity, $F^e(k' + N/2) = F^e(k')$, $F^o(k' + N/2) = F^o(k')$ for any $k' = 0, 1, \dots, N/2 - 1$.



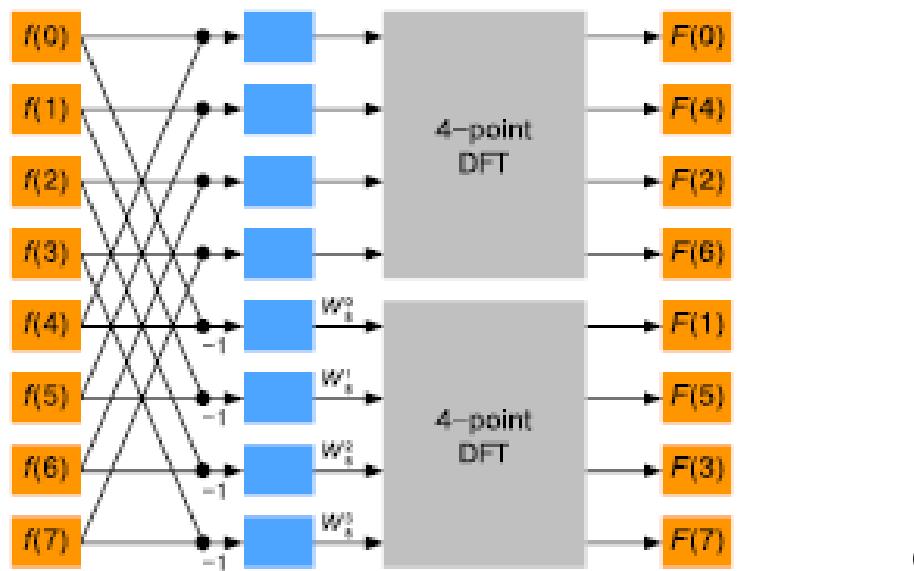
Courtesy: Pavel Karas.

Decimation in frequency (DIF)

- The input sequence f of the length N is divided into sequences f^r and f^s as $f^r(n') = f(n') + f(n' * N/2)$, $f^s = (f(n') - f(n' + N/2)) W^{n'N}$.
- Their Fourier transform fulfills: $F^r(k') = F(2k')$ and $F^s(k') = 2k' + 1$ for any $k' = 0, 1, \dots, N/2 - 1$.
- Sequences f^r and f^s can be processed recursively with inverse equations

$$f(n') = \frac{1}{2} \left(f^r(n') + f^s(n') W^{N-n'} \right),$$

$$f(n' + \frac{N}{2}) = \frac{1}{2} \left(f^r(n') - f^s(n') W^{N-n'} \right).$$



Courtesy: Pavel Karas.

FFT, decimation in time, the proof idea

$$\begin{aligned}
 F(k) &= \sum_{n=0}^{N-1} e^{\frac{-2\pi i k n}{N}} f(n) \\
 &= \sum_{n=0}^{(N/2)-1} e^{\frac{-2\pi i k (2n)}{N}} f(2n) + \sum_{n=0}^{(N/2)-1} e^{\frac{-2\pi i k (2n+1)}{N}} f(2n+1) \\
 &= \sum_{n=0}^{(N/2)-1} e^{\frac{-2\pi i k n}{N/2}} f(2n) + W^n \sum_{n=0}^{(N/2)-1} e^{\frac{-2\pi i k n}{N/2}} f(2n+1) \\
 &= F^e(k) + W^n F^o(k), \quad k = 1, \dots, N
 \end{aligned}$$

- ◆ The key idea: recursiveness and N is power of 2.
- ◆ Only $\log_2 N$ iterations needed.

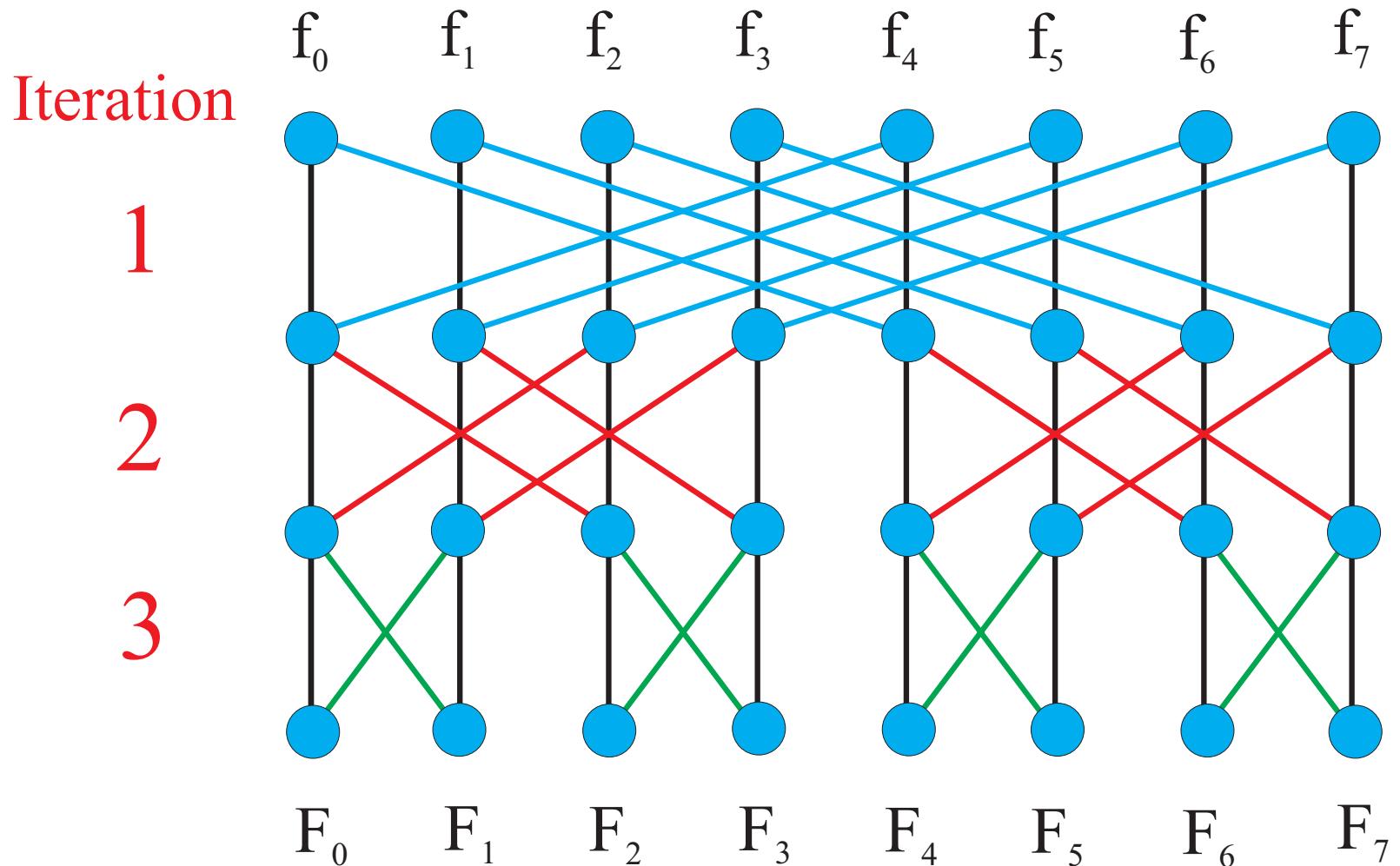
FFT, the proof idea (2)

- ◆ Spectra $F^e(k)$ and $F^o(k)$ are periodic in k with length $N/2$.
- ◆ What is Fourier transform o length 1? Answer: It is just identity.
- ◆ For every pattern of $\log_2 N$ e's and o's, there is a one-point transform that is just one of input numbers $f(n)$,

$$F^{eoeeeoeo\dots oee}(k) = f(n) \quad \text{for some } n .$$

- ◆ The next trick is to utilize partial results \implies butterfly scheme of computations.

FFT butterfly calculation scheme



2D Fourier transform

The idea. The image function $f(x, y)$ is decomposed to a linear combination of harmonic (sines and cosines, more generally orthogonal) functions.

Definition of the direct transform. u, v are spatial frequencies.

$$F(u, v) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} f(x, y) e^{-2\pi i(xu+yv)} dx dy$$

Inverse Fourier transform

$$f(x, y) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} F(u, v) e^{2\pi i(xu+yv)} du dv$$

- ◆ $f(x, y)$ is a linear combination of simple harmonic functions (components)
 $e^{2\pi i(xu+uv)}$.
- ◆ Thanks to Euler formula
 in general $e^{iz} = \cos z + i \sin z$, here $\cos(-2\pi ixu) + i \sin(-2\pi ixu)$,
 \cos corresponds to the real part and \sin corresponds to the imaginary part.
- ◆ Function $F(u, v)$ (complex spectrum) gives weights of harmonic components in the linear combination.

Spatial frequencies spectrum

The outcome of the Fourier transform is a complex function $F(u, v)$.

(Complex) spectrum
$$F(u, v) = F_{Re}(u, v) + i F_{Im}(u, v)$$

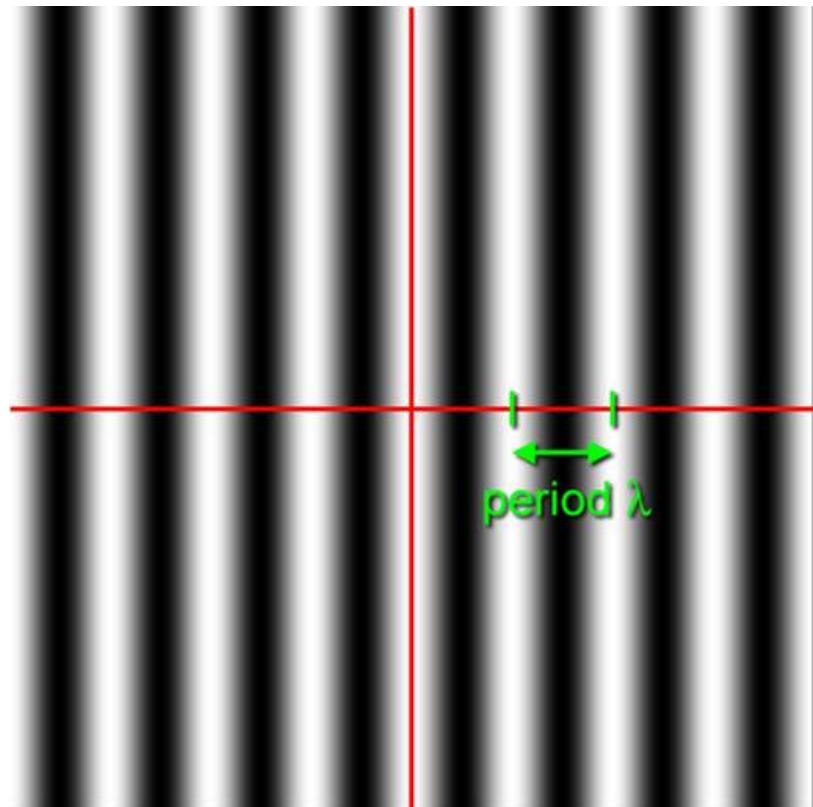
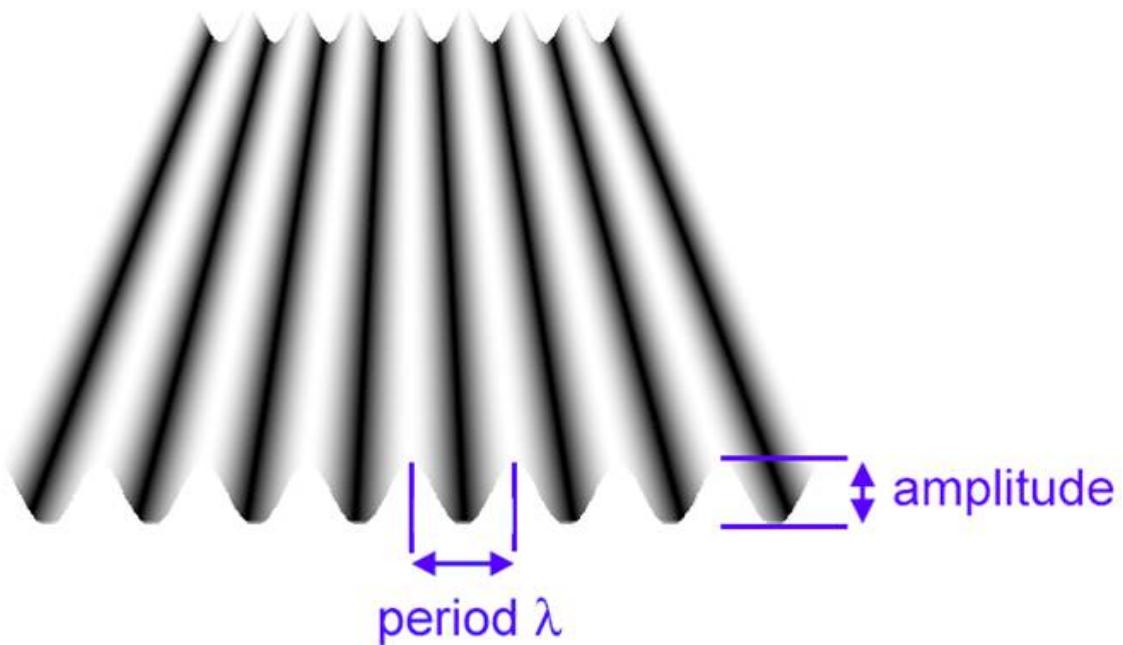
Amplitude spectrum
$$|F(u, v)| = \sqrt{F_{Re}^2(u, v) + F_{Im}^2(u, v)}$$

Phase angle spectrum
$$\phi(u, v) = \tan^{-1} \left[\frac{F_{Im}(u, v)}{F_{Re}(u, v)} \right]$$

Power spectrum
$$P(u, v) = |F(u, v)|^2 = F_{Re}^2(u, v) + F_{Im}^2(u, v)$$

2D sinusoid, illustration

- ◆ 2D sinusoids can be imagined as plane waves with the amplitude shown as intensity (gray level).
- ◆ The analogy to corrugated iron comes from a topographic displaying of a 2D sinusoid (or cosinusoid).



2D sinusoid, illustration (2)

$$\omega = \sqrt{u^2 + v^2}, \quad u = \omega \cos \Theta, \quad v = \omega \sin \Theta, \quad \Theta = \tan^{-1} \left(\frac{v}{u} \right).$$

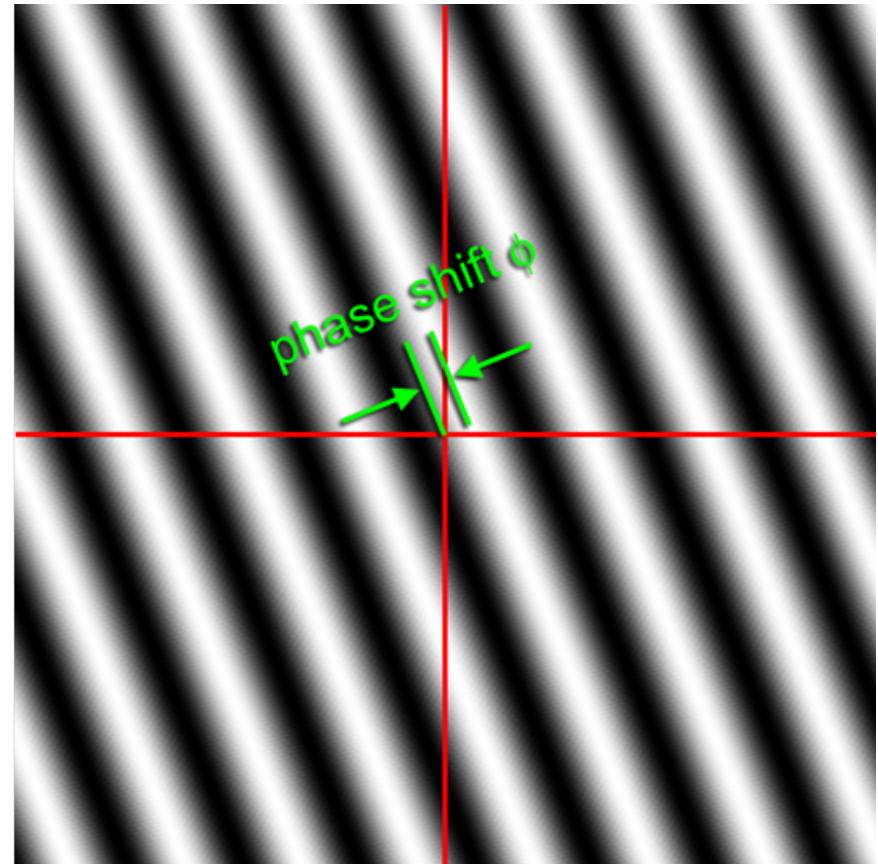
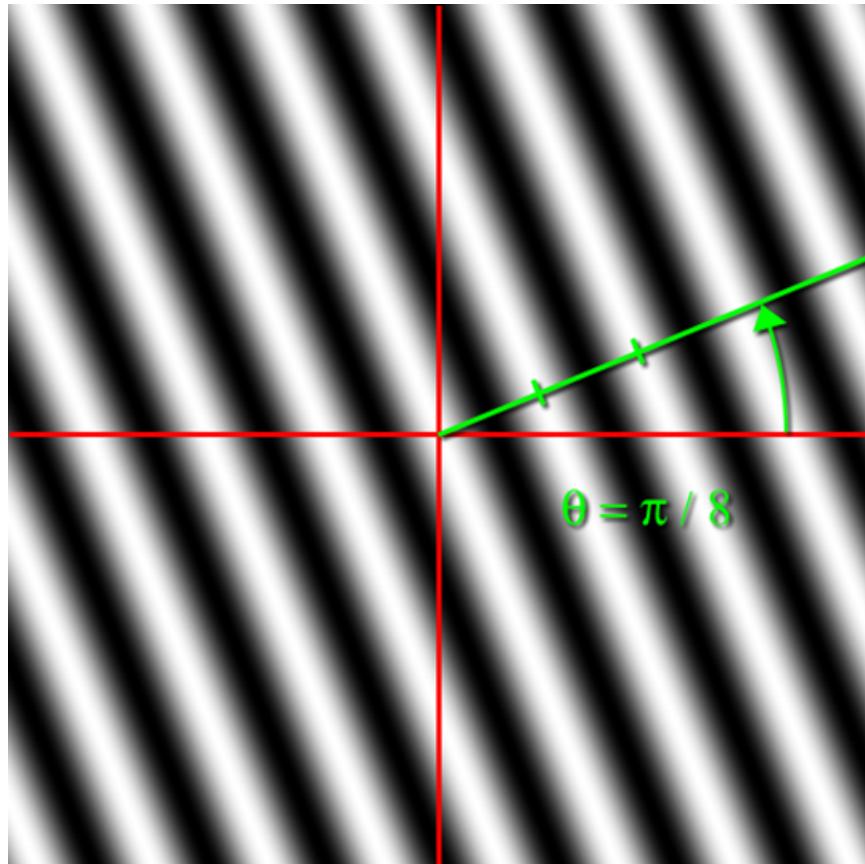
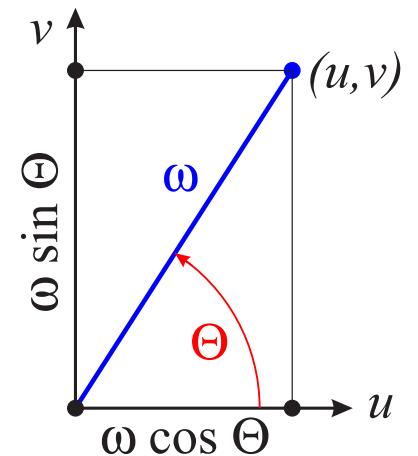
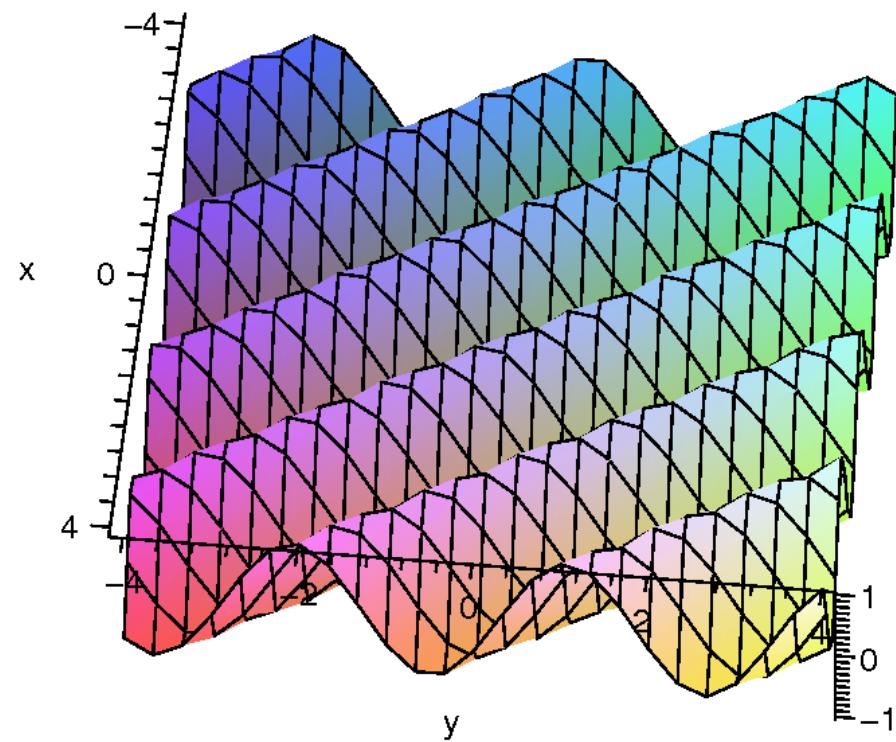
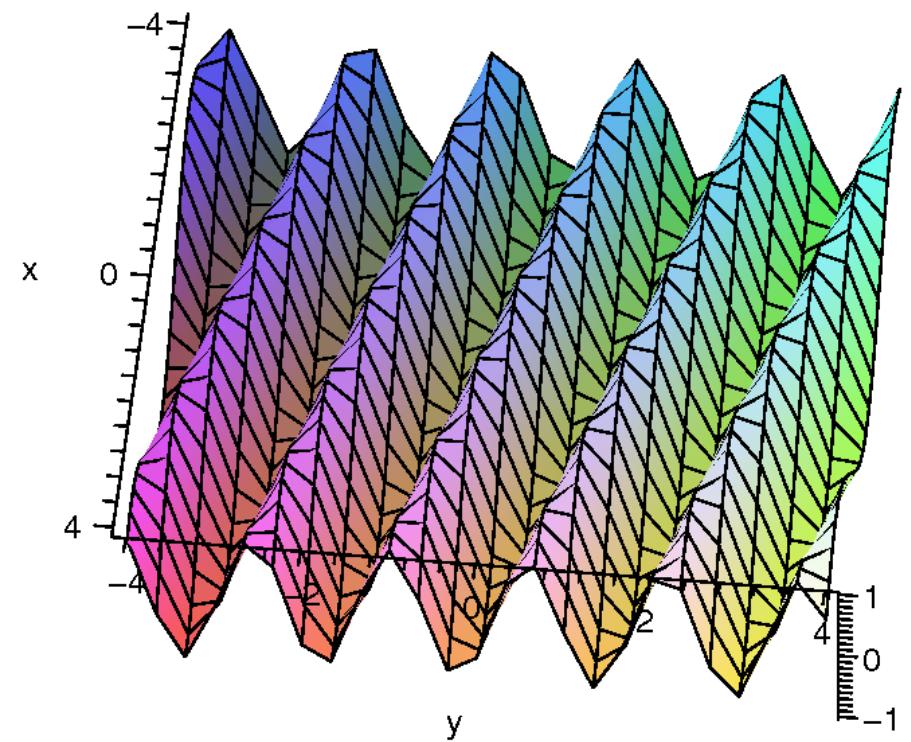


Illustration of 2D FT bases vectors

Analogy – corrugated iron.



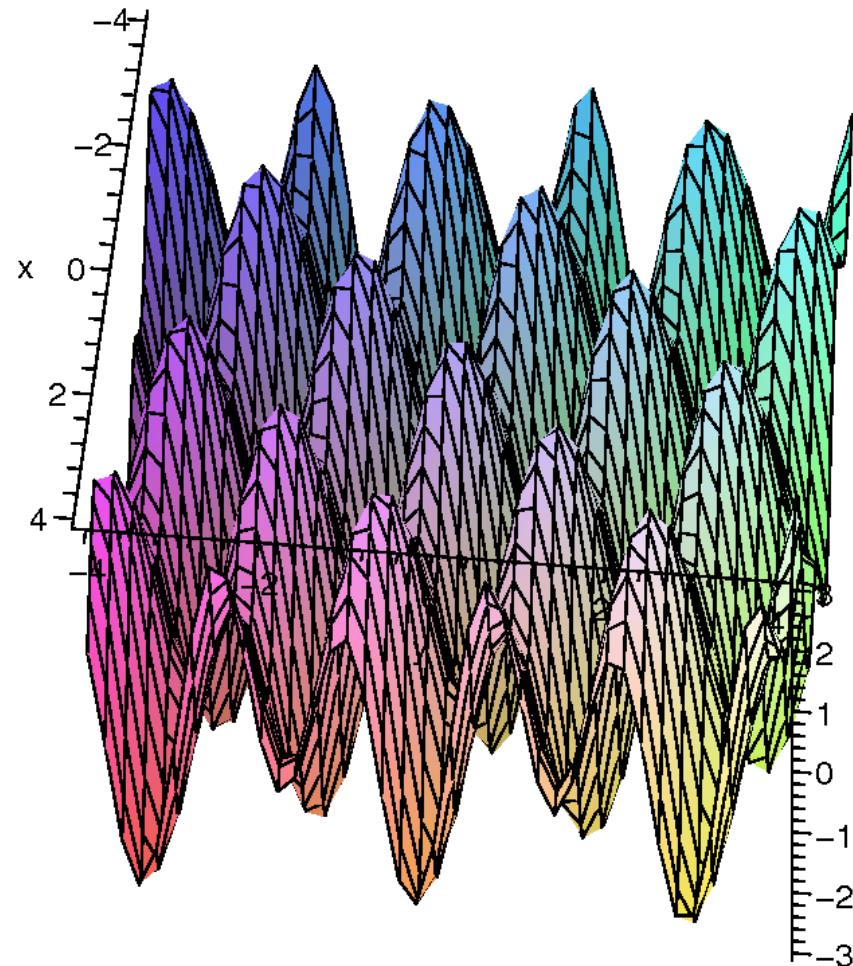
$$\sin(3x + 2y)$$



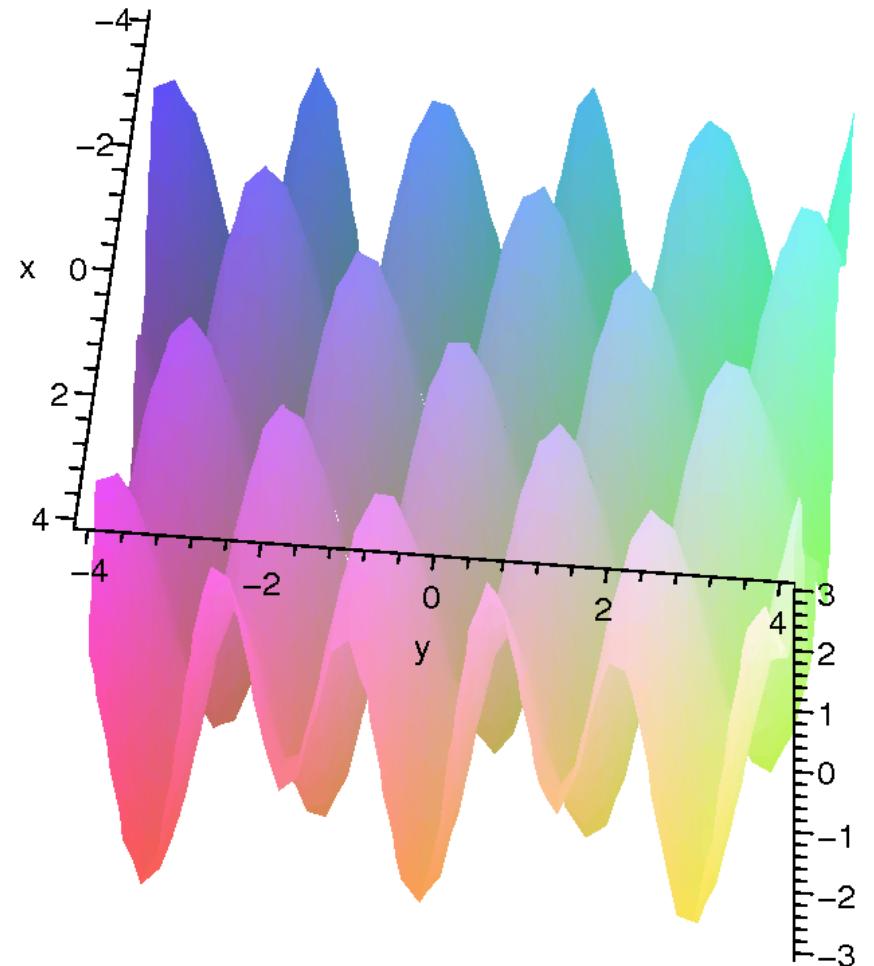
$$\cos(x + 4y)$$

Linear combination of base vectors

Analogy – carton egg tray.



$$\sin(3x + 2y) + \cos(x + 4y)$$



different display only

Carton egg tray



2D discrete Fourier transform

Direct transform

$$F(u, v) = \frac{1}{MN} \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} f(m, n) \exp \left[-2\pi i \left(\frac{mu}{M} + \frac{nv}{N} \right) \right] ,$$

$$u = 0, 1, \dots, M - 1 , \quad v = 0, 1, \dots, N - 1 ,$$

Inverse transform

$$f(m, n) = \sum_{u=0}^{M-1} \sum_{v=0}^{N-1} F(u, v) \exp \left[2\pi i \left(\frac{mu}{M} + \frac{nv}{N} \right) \right] ,$$

$$m = 0, 1, \dots, M - 1 , \quad n = 0, 1, \dots, N - 1 .$$

2D Fourier Tx as twice 1D Fourier Tx

2D direct FT can be modified to

$$F(u, v) = \frac{1}{M} \sum_{m=0}^{M-1} \left[\frac{1}{N} \sum_{n=0}^{N-1} \exp\left(\frac{-2\pi i nv}{N}\right) f(m, n) \right] \exp\left(\frac{-2\pi i mu}{M}\right),$$

$$u = 0, 1, \dots, M - 1, \quad v = 0, 1, \dots, N - 1.$$

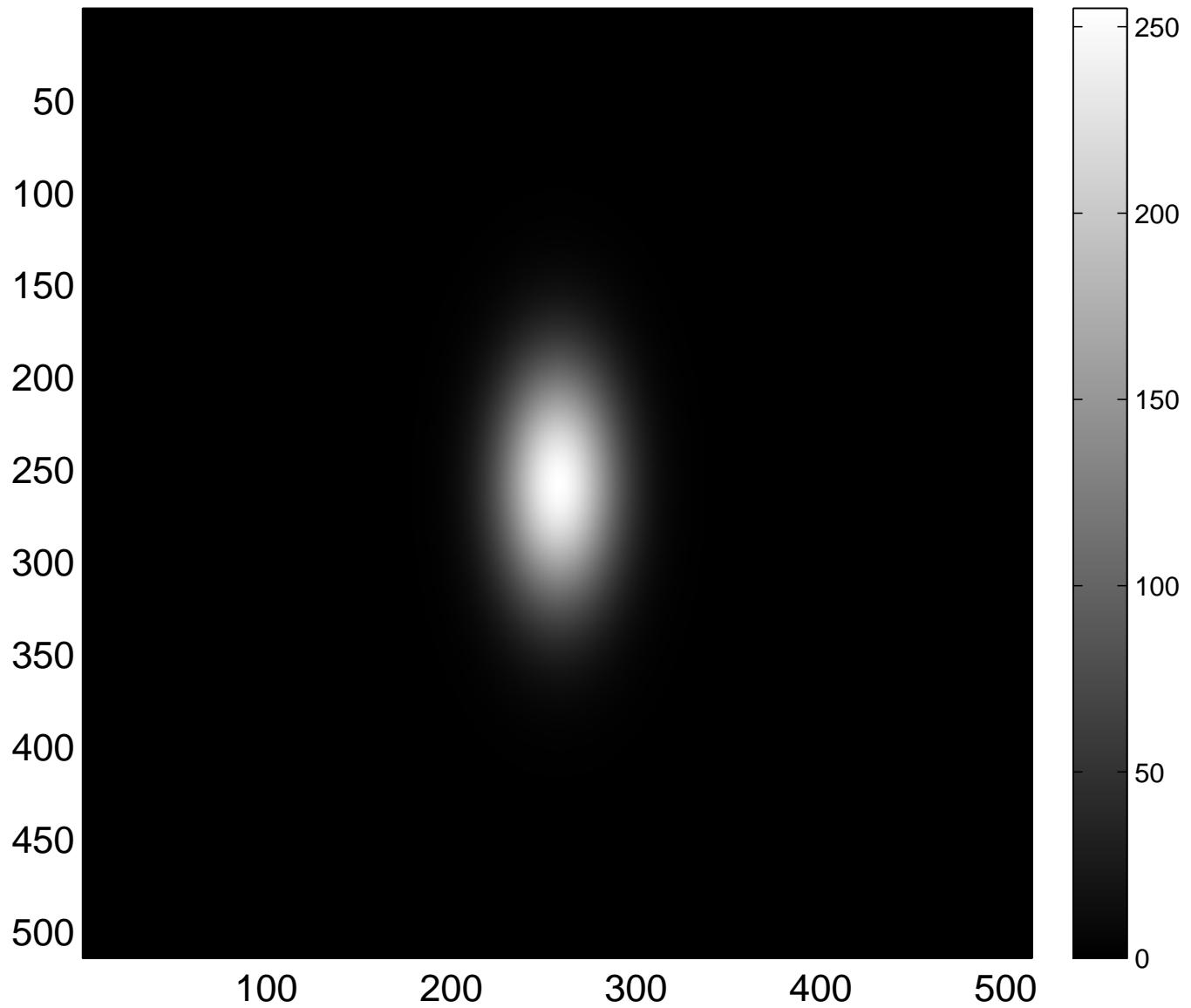
- ◆ The term in square brackets corresponds to the one-dimensional Fourier transform of the m^{th} line and can be computed using the standard fast Fourier transform (FFT).
- ◆ Each line is substituted with its Fourier transform, and the one-dimensional discrete Fourier transform of each column is computed.

Displaying spectra, 2D Gaussian example

Gaussian is selected for illustration because it has a smooth spectrum, cf. uncertainty principle.

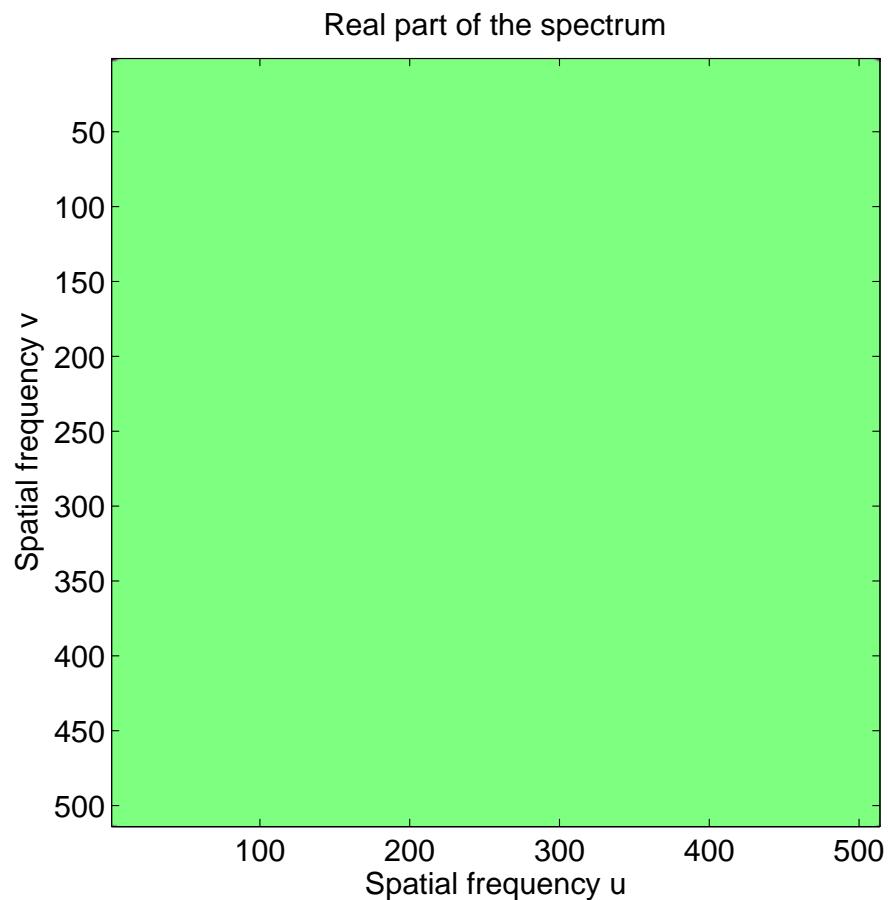


Input intensity image, coordinate system

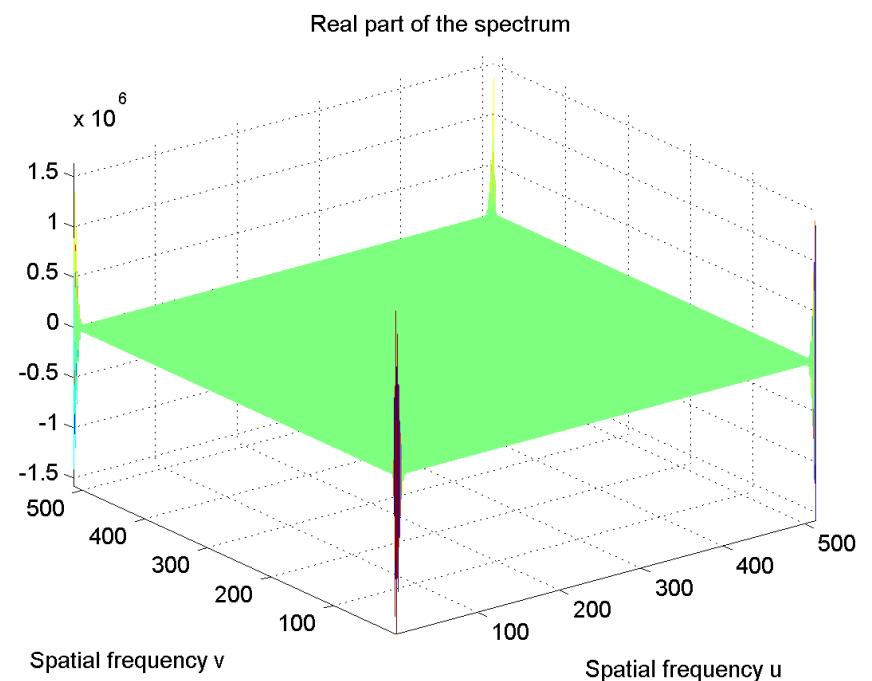


Real part of the spectrum, image and mesh

Problem with the image related coordinate system related to the image: interesting information is in corners, moreover divided into quarters. Due to spectrum periodicity it can be arbitrarily shifted.

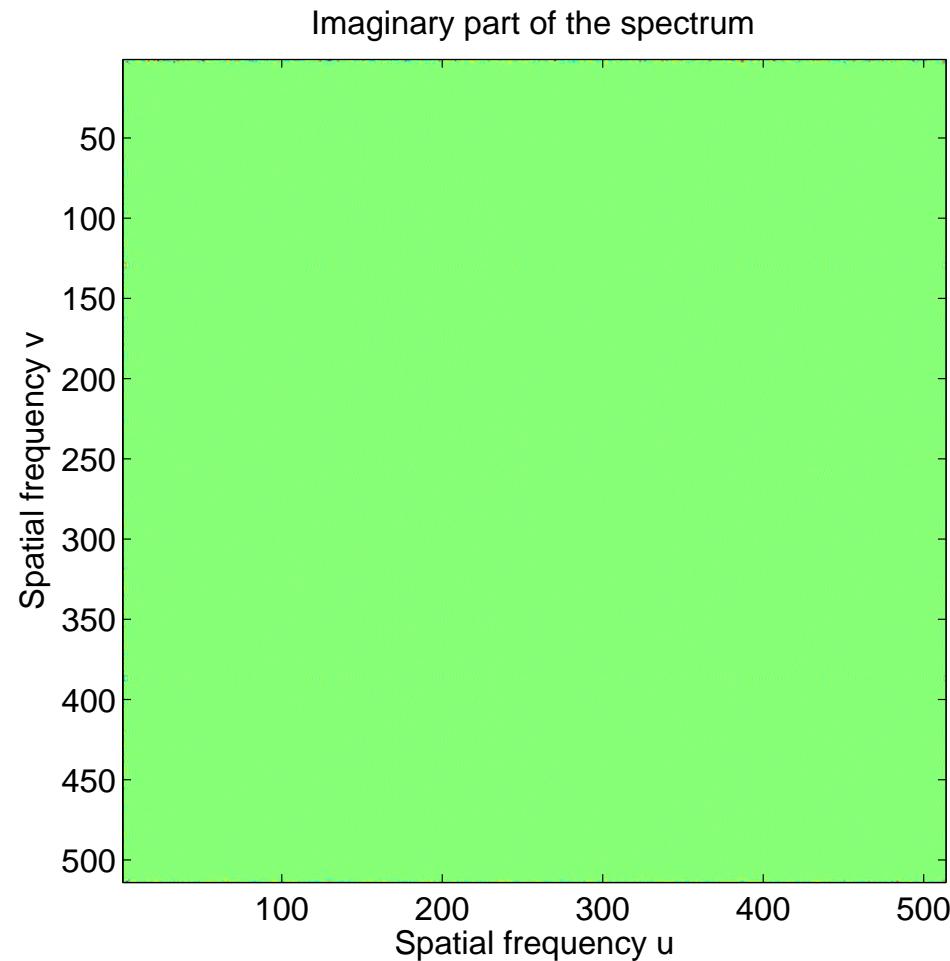


real part, image

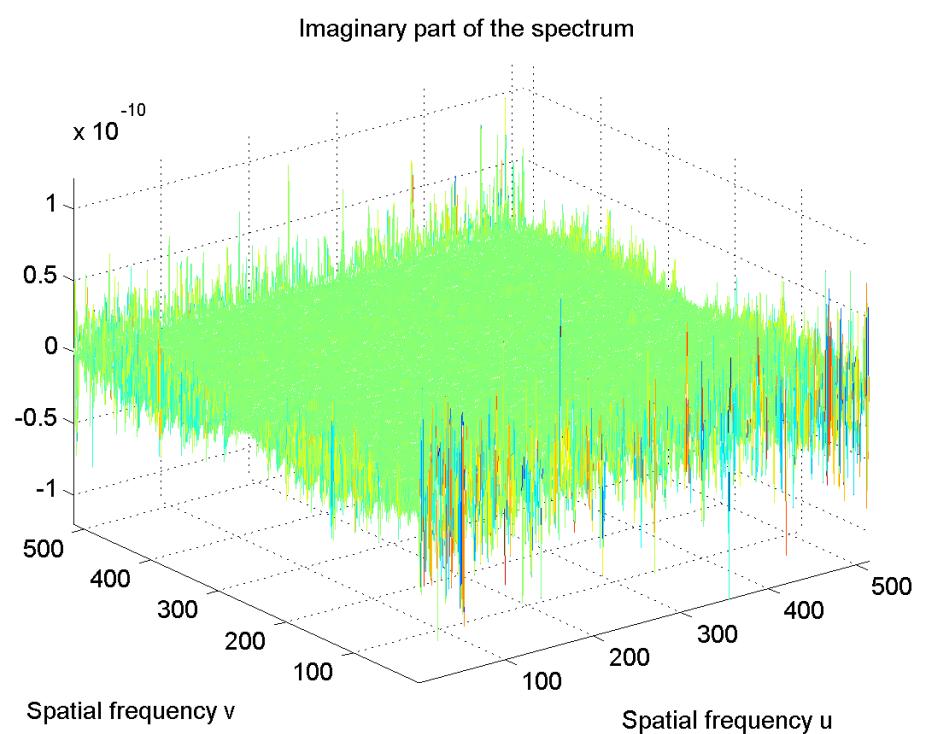


real part, mesh

Imaginary part of the spectrum, image and mesh

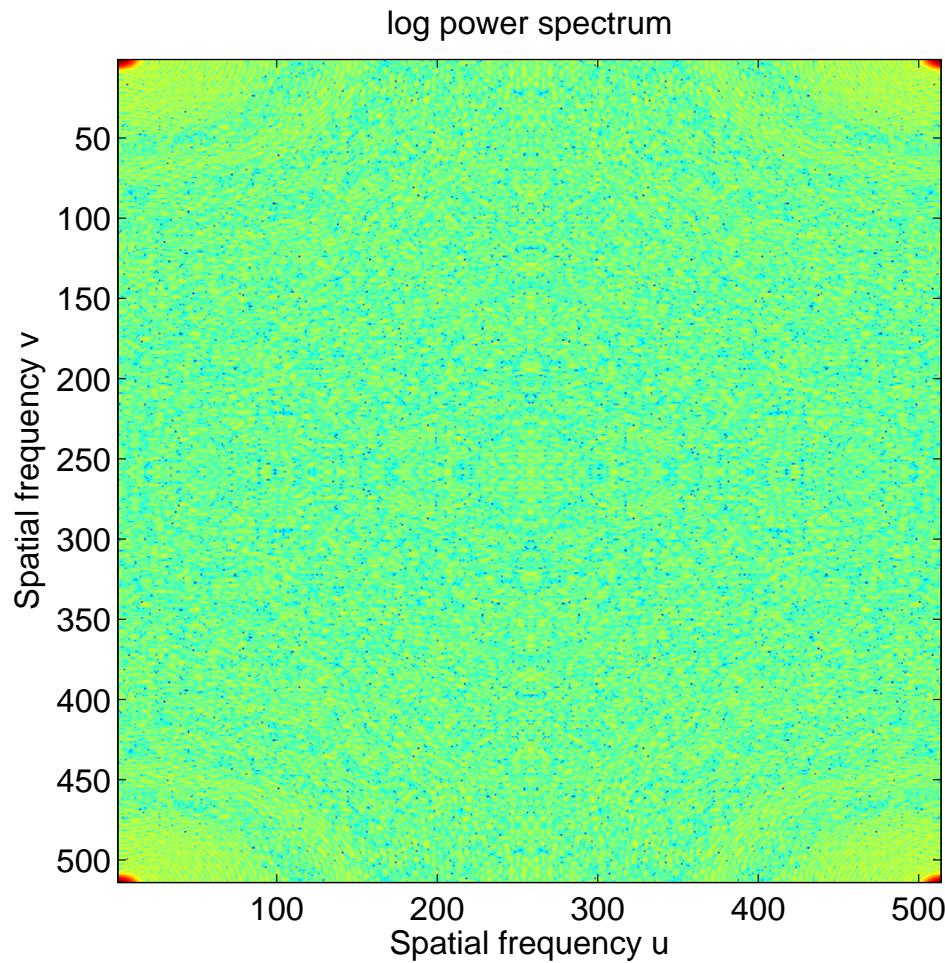


imaginary part, image

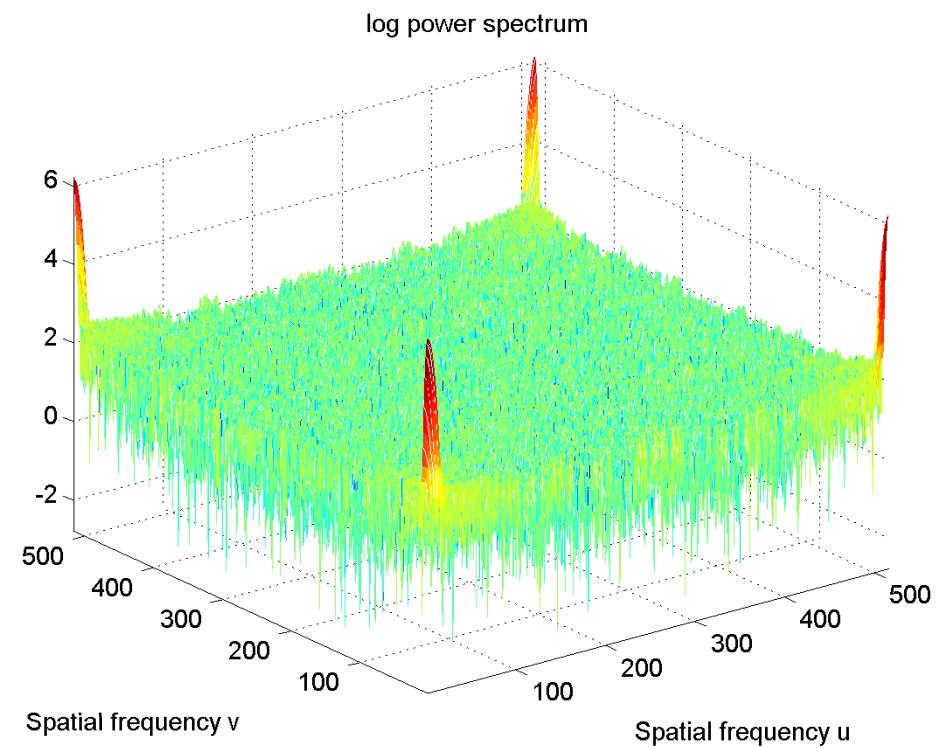


imaginary part, mesh

Log power of the spectrum, image and mesh



image

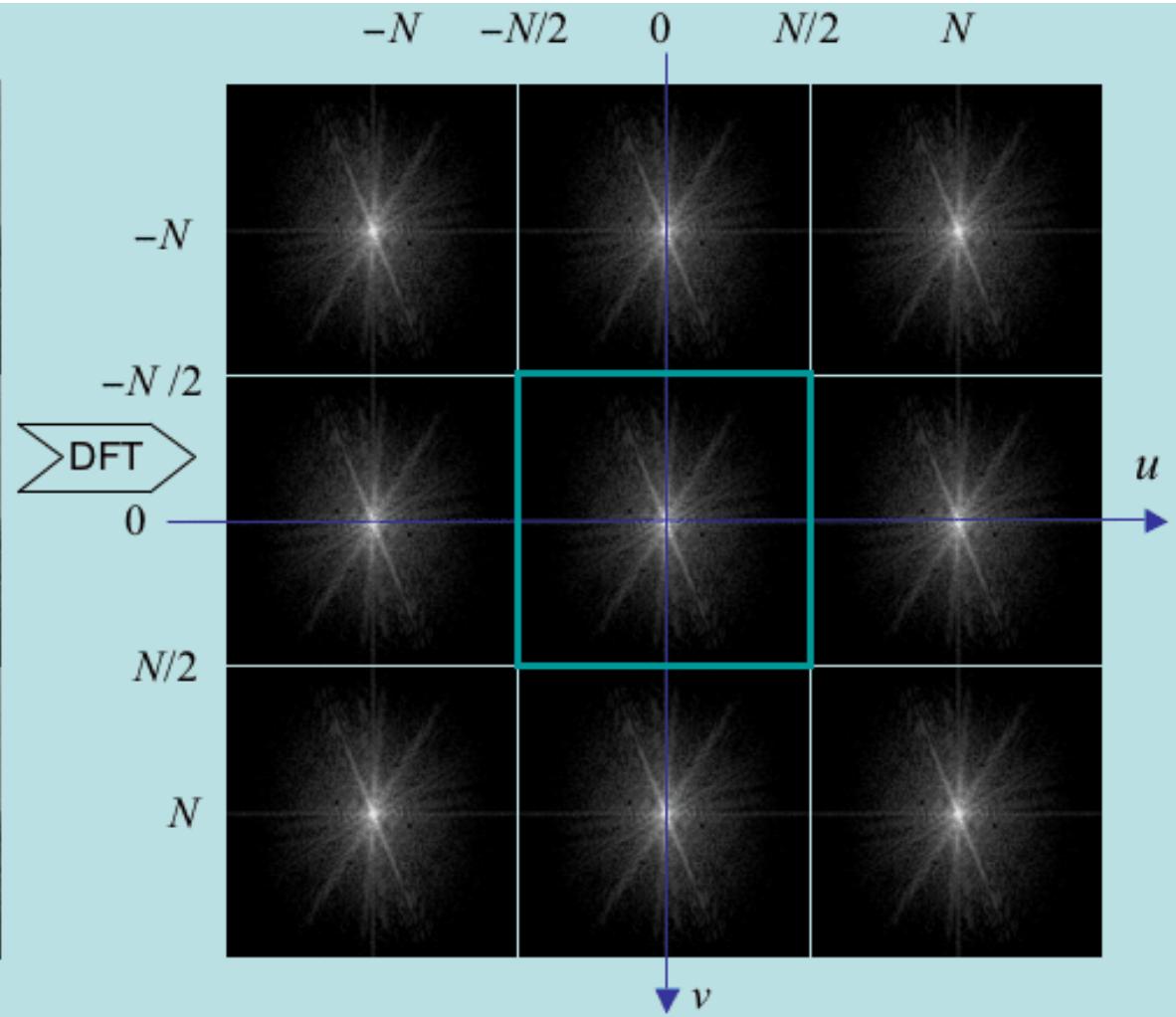


mesh

Periodic image



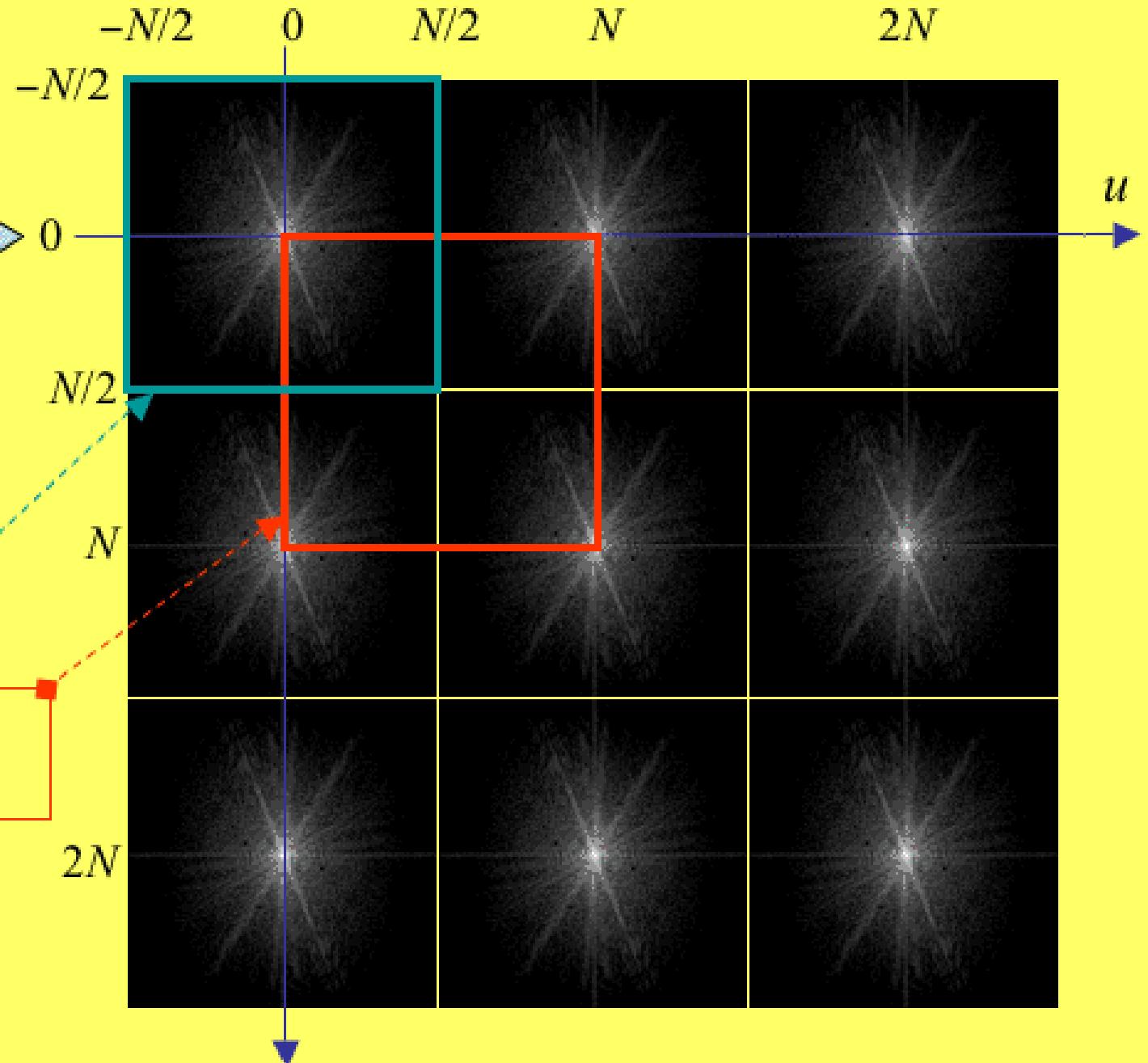
Spatial discontinuities caused by considering an image to be periodic



Periodic spectrum

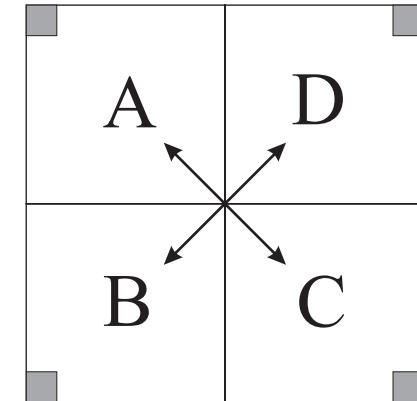


DFT

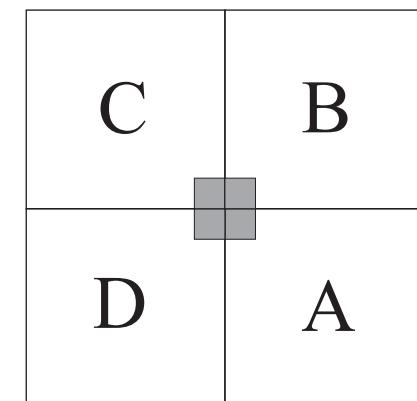


Centered spectra

- ◆ It is useful to visualize a centered spectrum with the origin of the coordinate system $(0, 0)$ in the middle of the spectrum.
- ◆ Assume the original spectrum is divided into four quadrants. The small gray-filled squares in the corners represent positions of low frequencies.
- ◆ Due to the symmetries of the spectrum the quadrant positions can be swapped diagonally and the low frequencies locations appear in the middle of the image.
- ◆ MATLAB provides function `fftshift` which converts noncentered \longleftrightarrow centered spectra by switching quadrants diagonally.

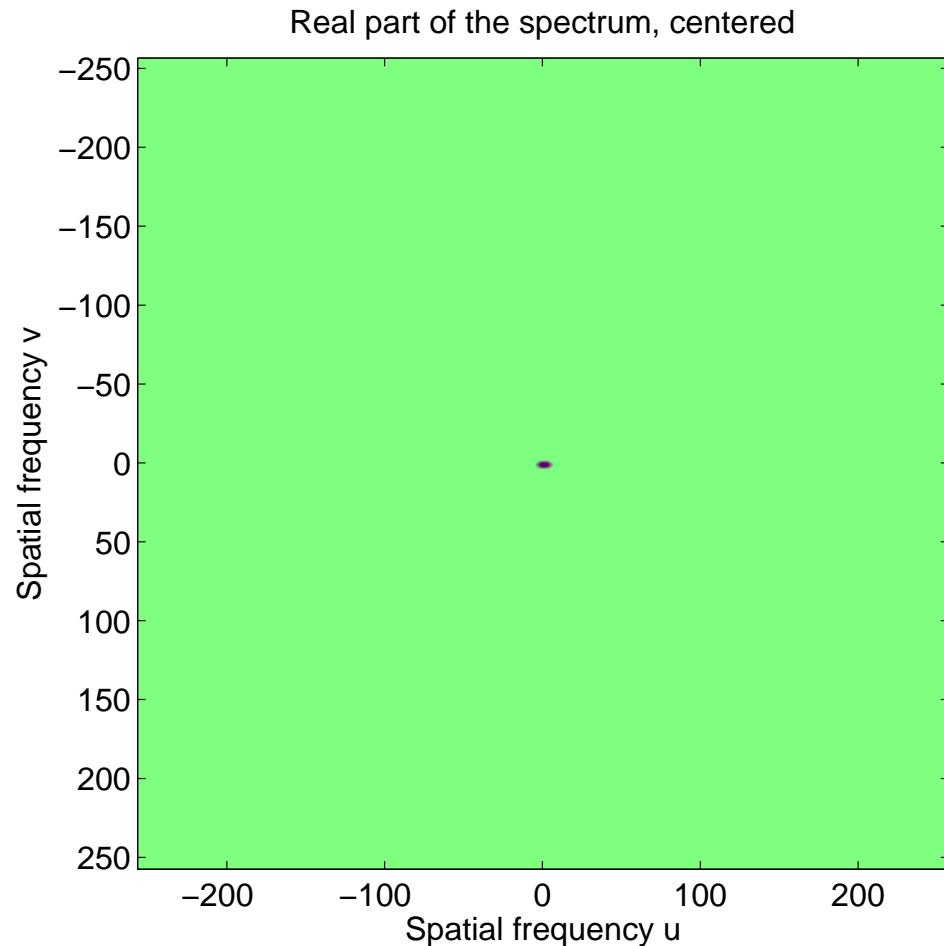


original spectrum
low frequencies in corners

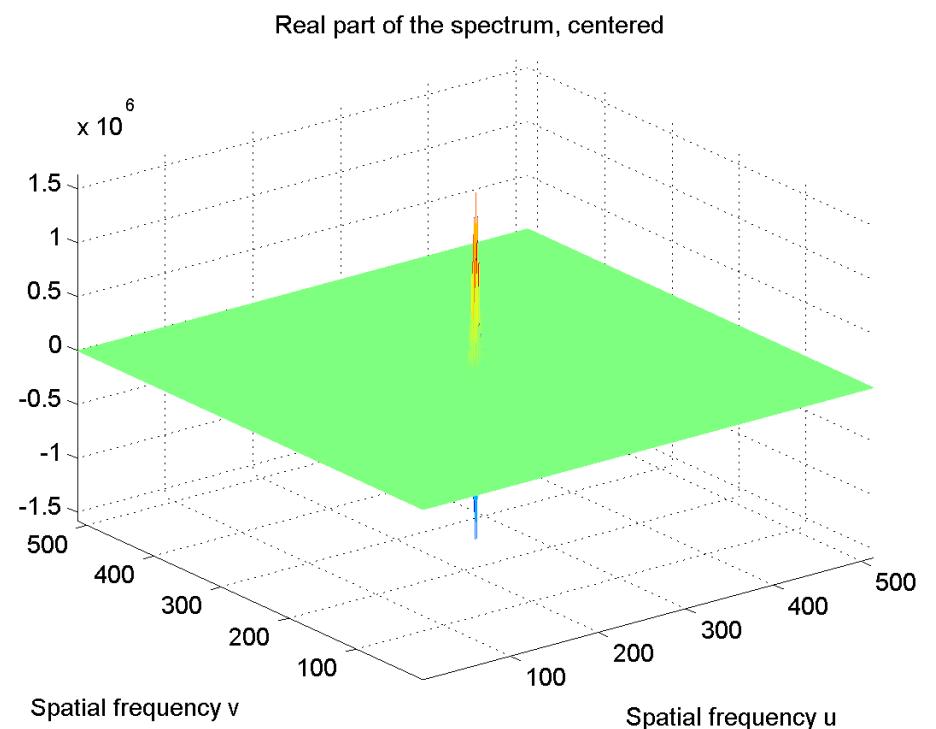


shifted spectrum
with the origin at $(0, 0)$

Real part of the centered spectrum, image and mesh

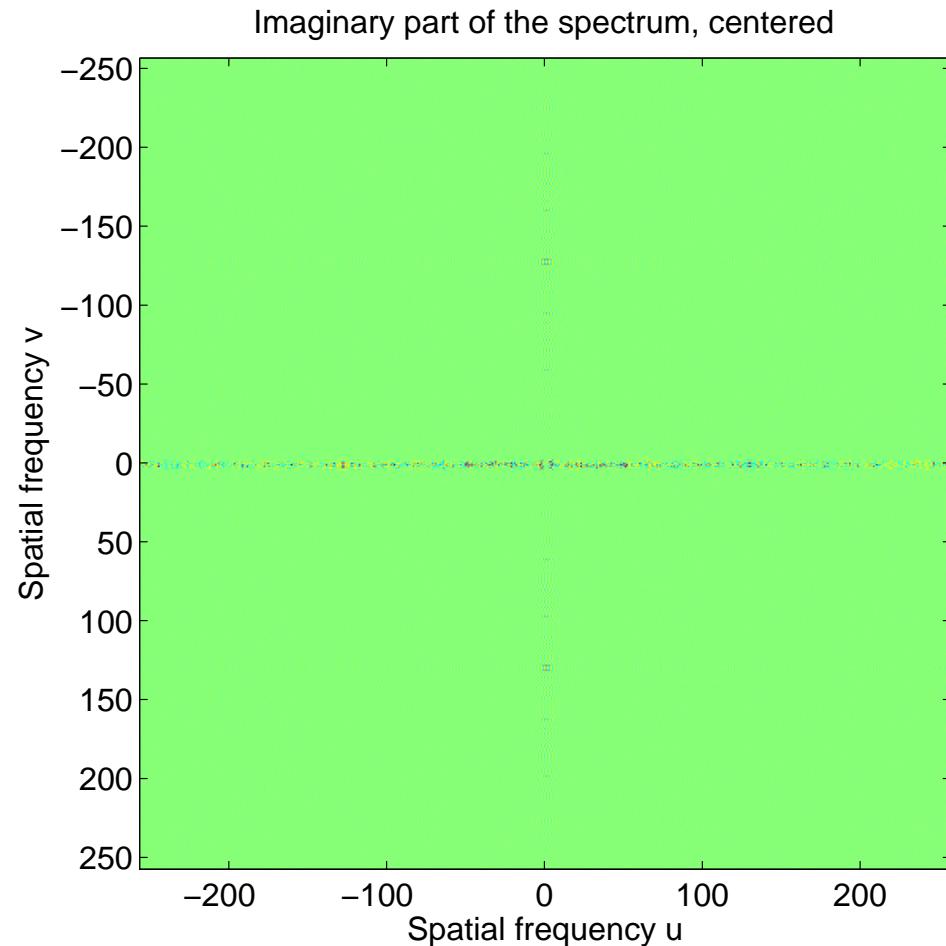


real part, image

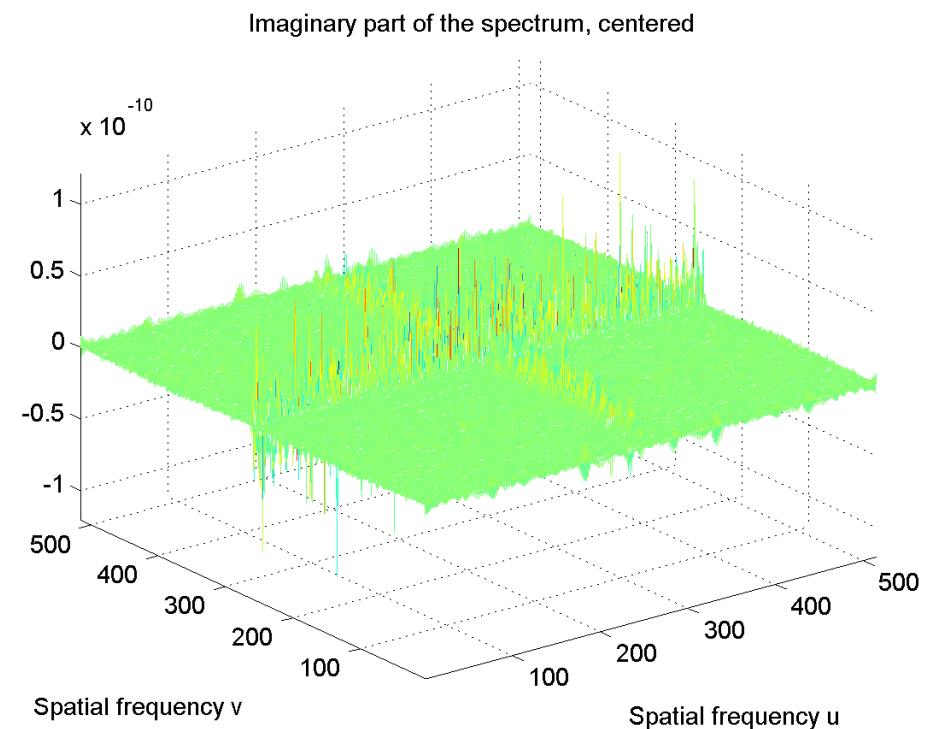


real part, mesh

Imaginary part of the centered spectrum image and mesh

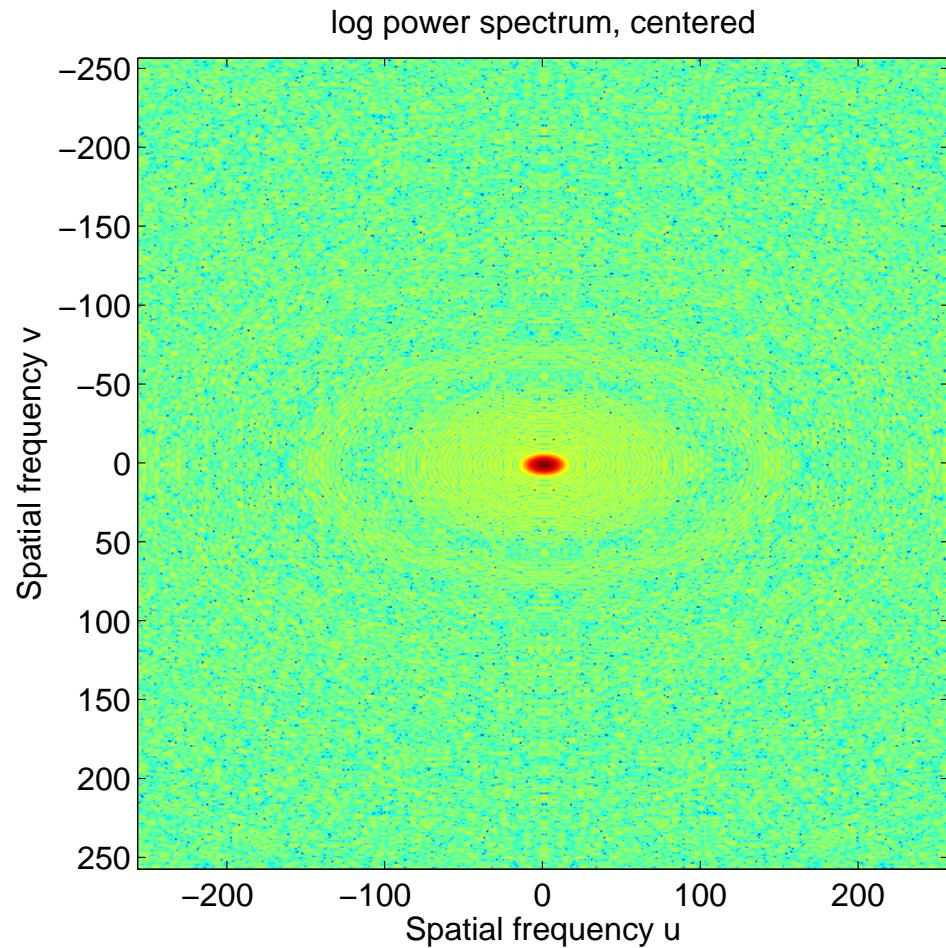


imaginary part, image

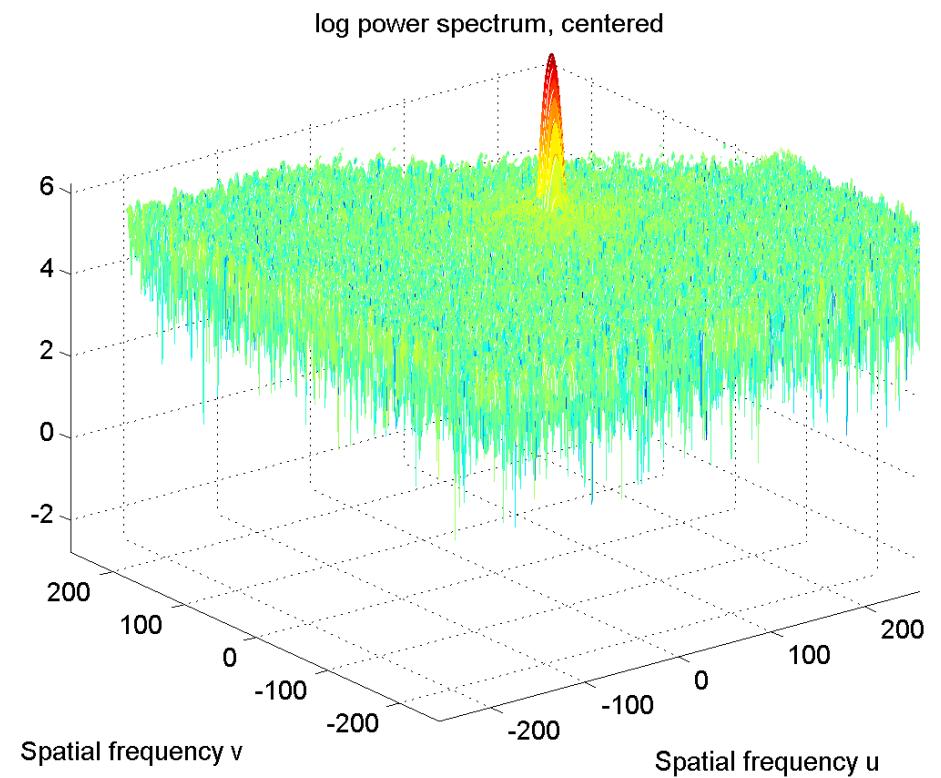


imaginary part, mesh

Log power of the centered spectrum image and mesh

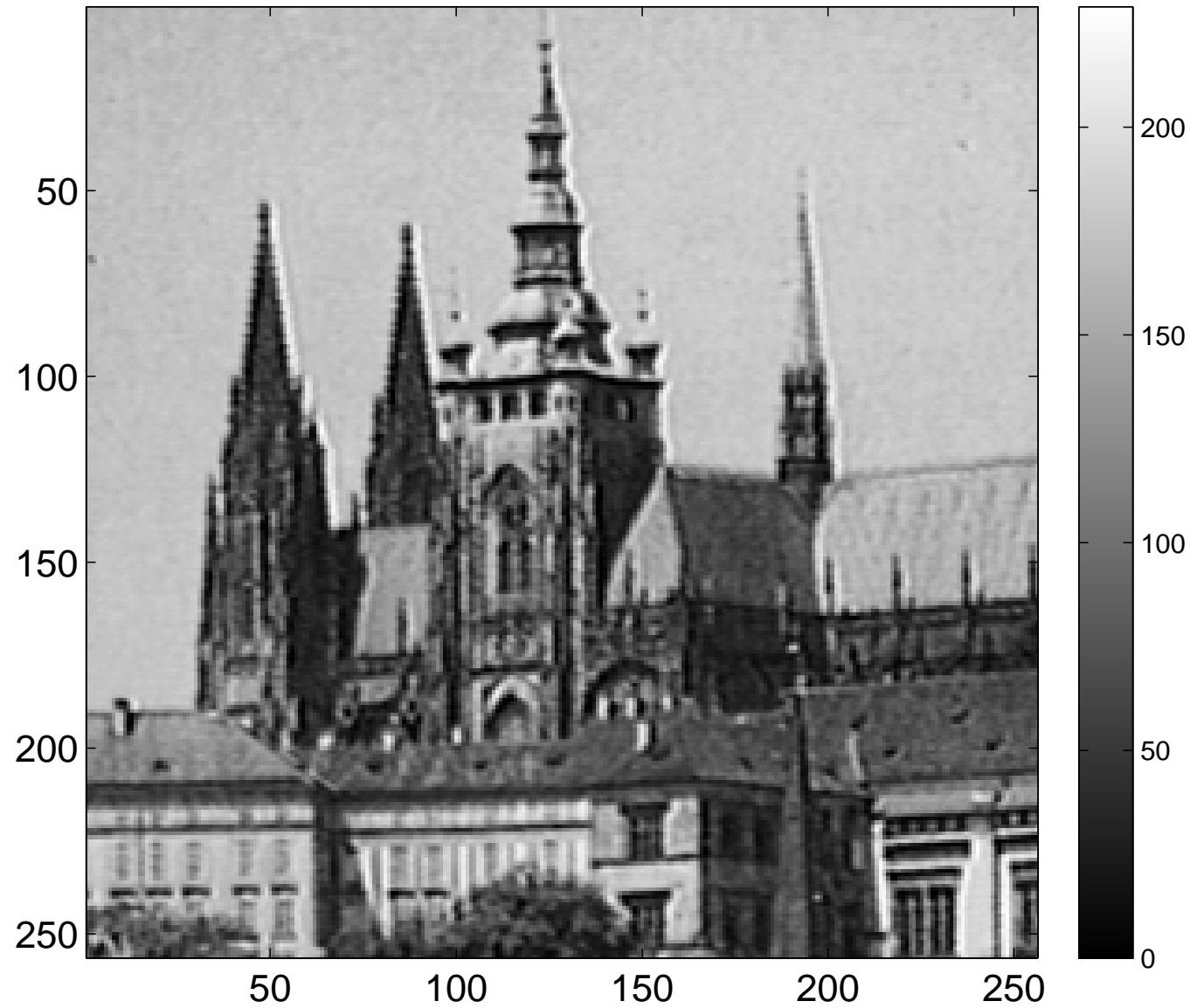


image

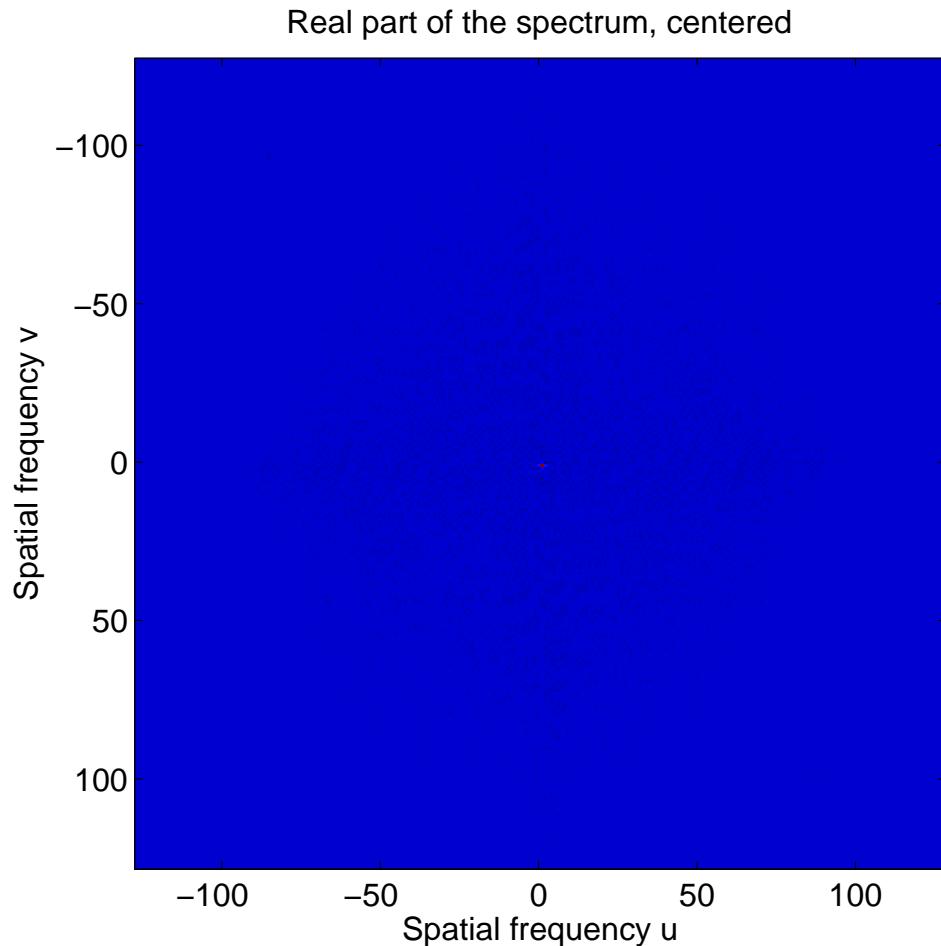


mesh

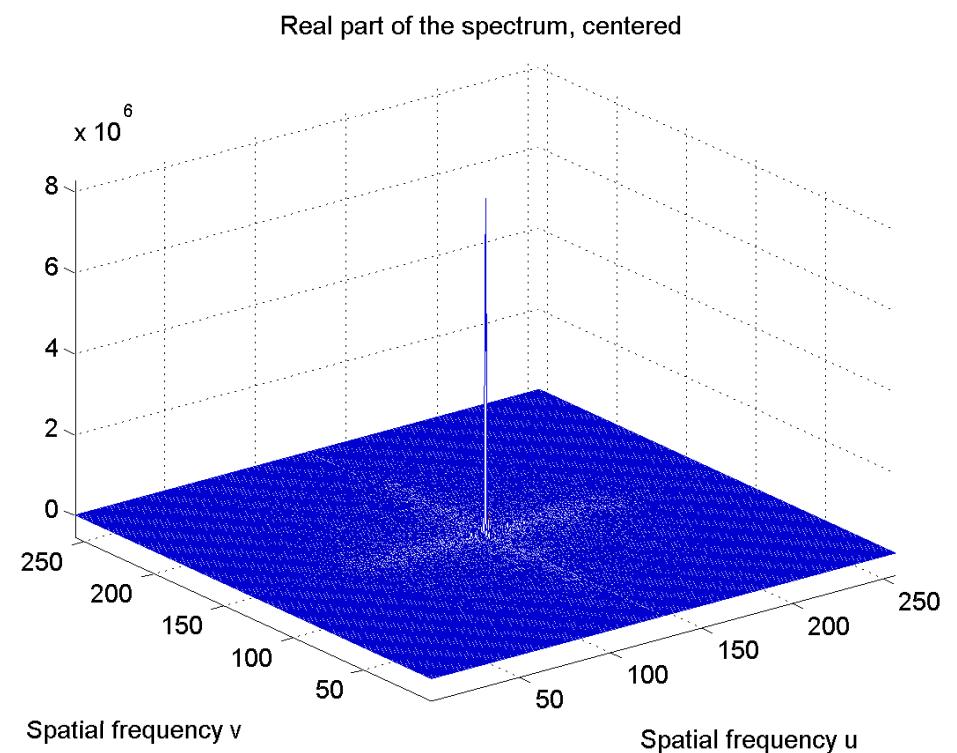
Prague Castle example, input image 265×256



Real part of the centered spectrum, image and mesh

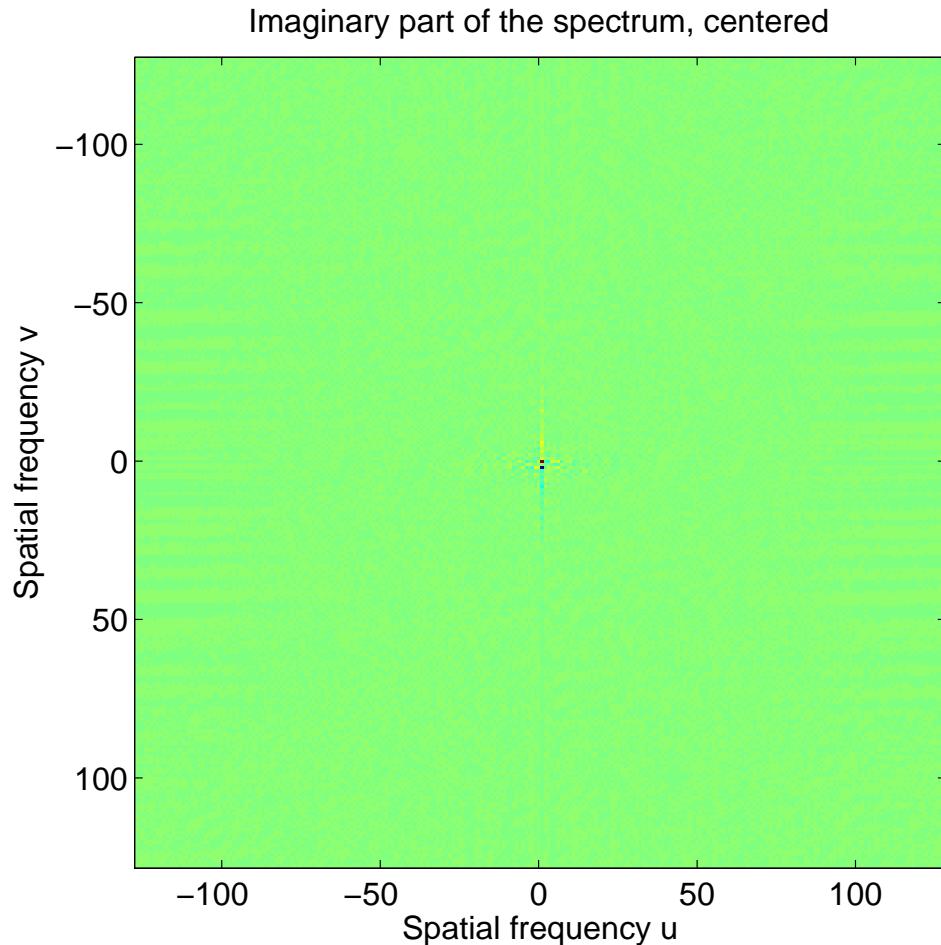


real part, image

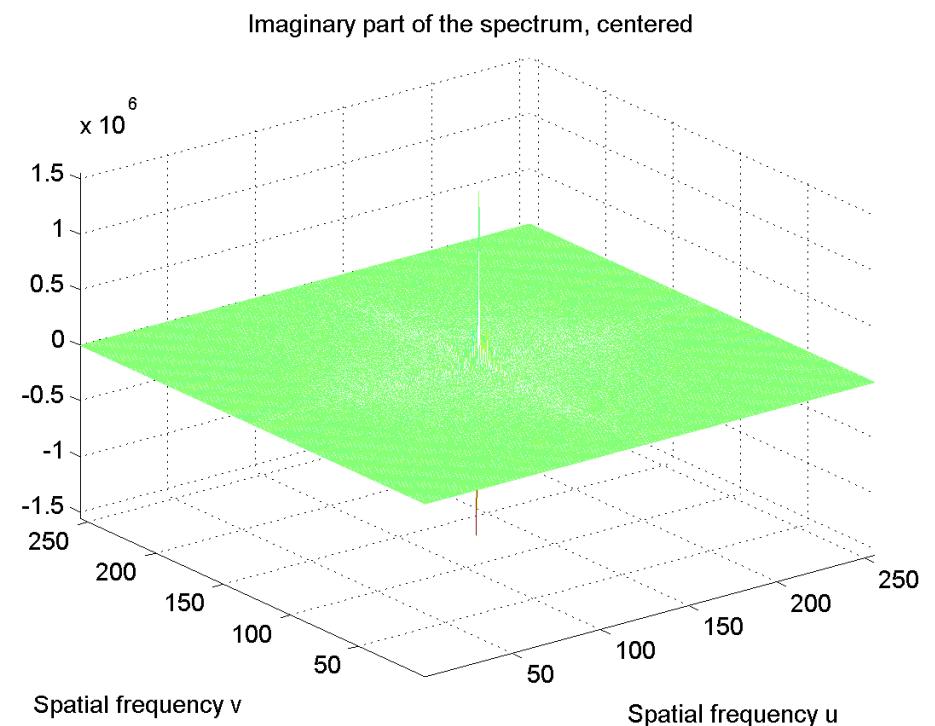


real part, mesh

Imaginary part of the centered spectrum image and mesh

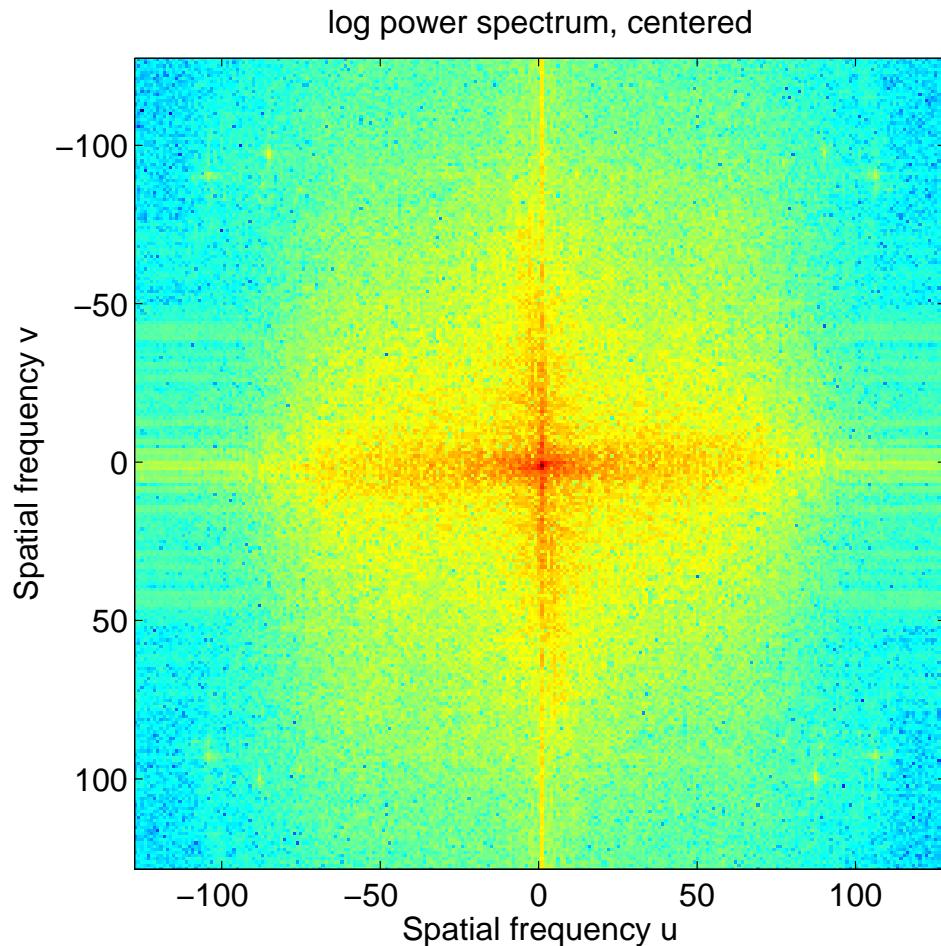


imaginary part, image

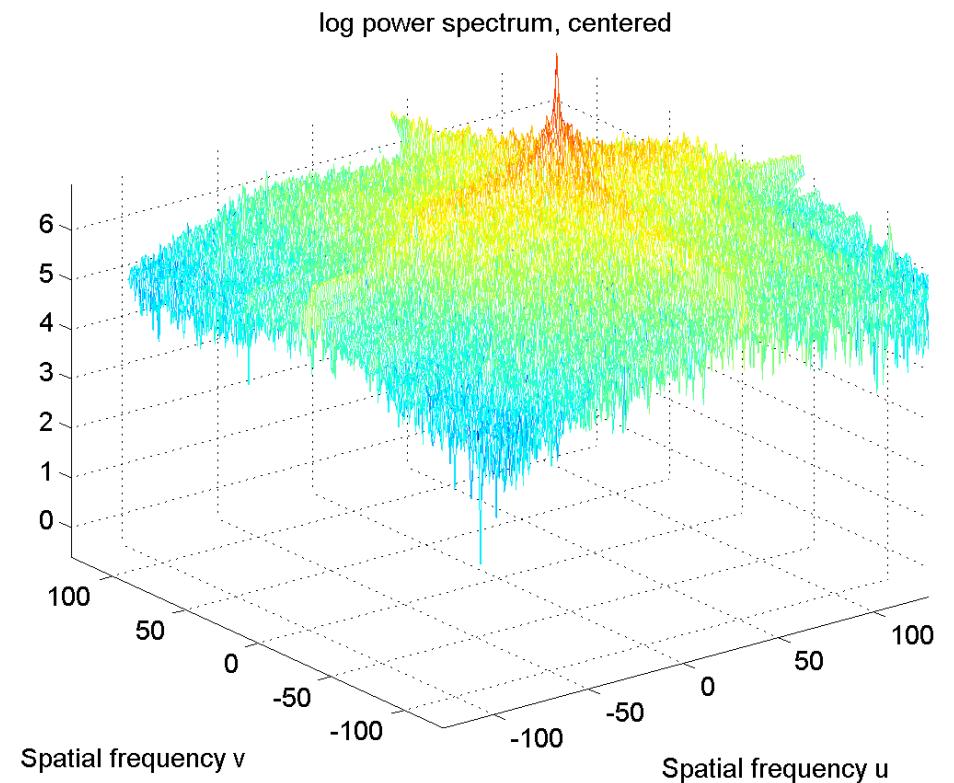


imaginary part, mesh

Log power of the centered spectrum image and mesh

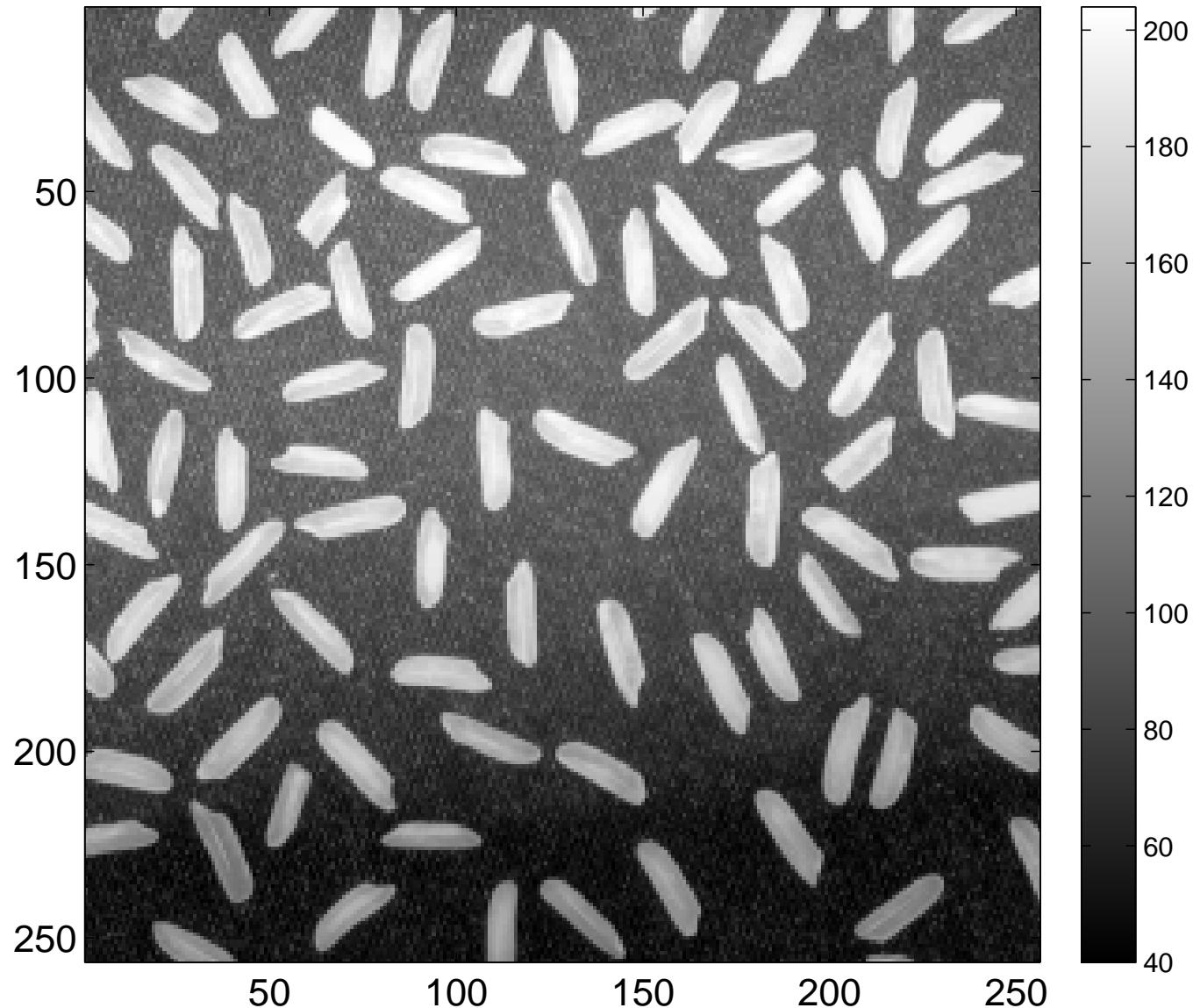


image

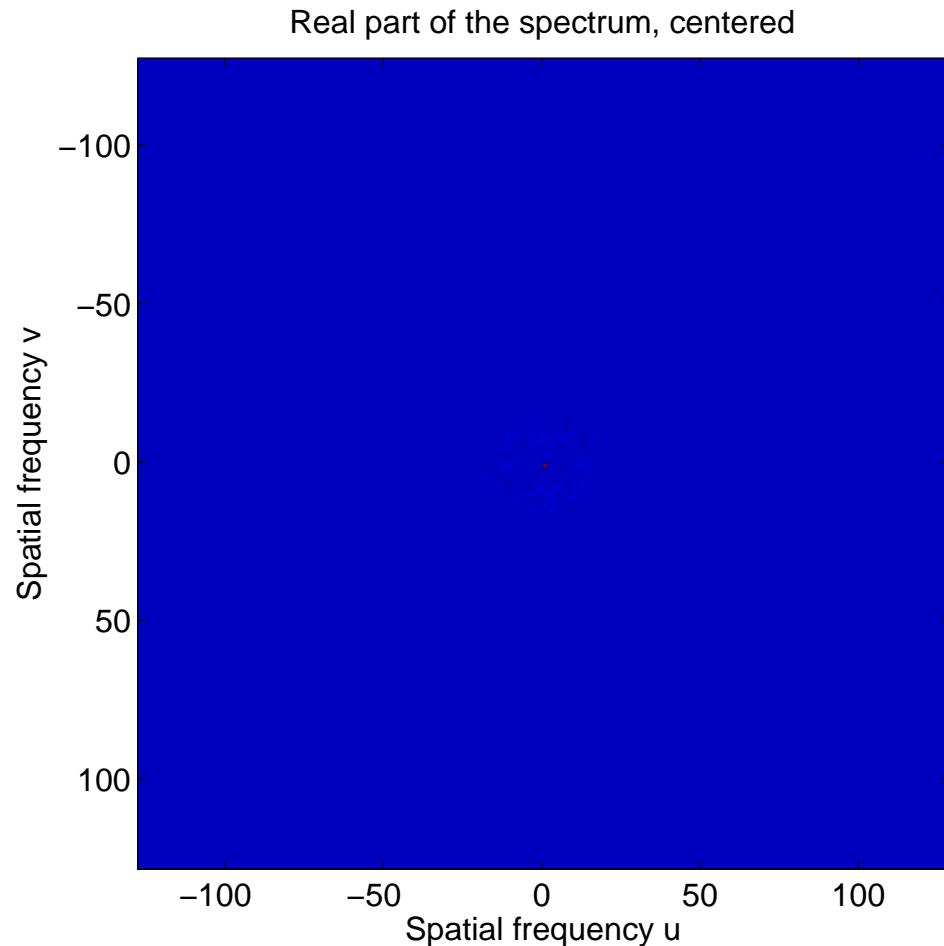


mesh

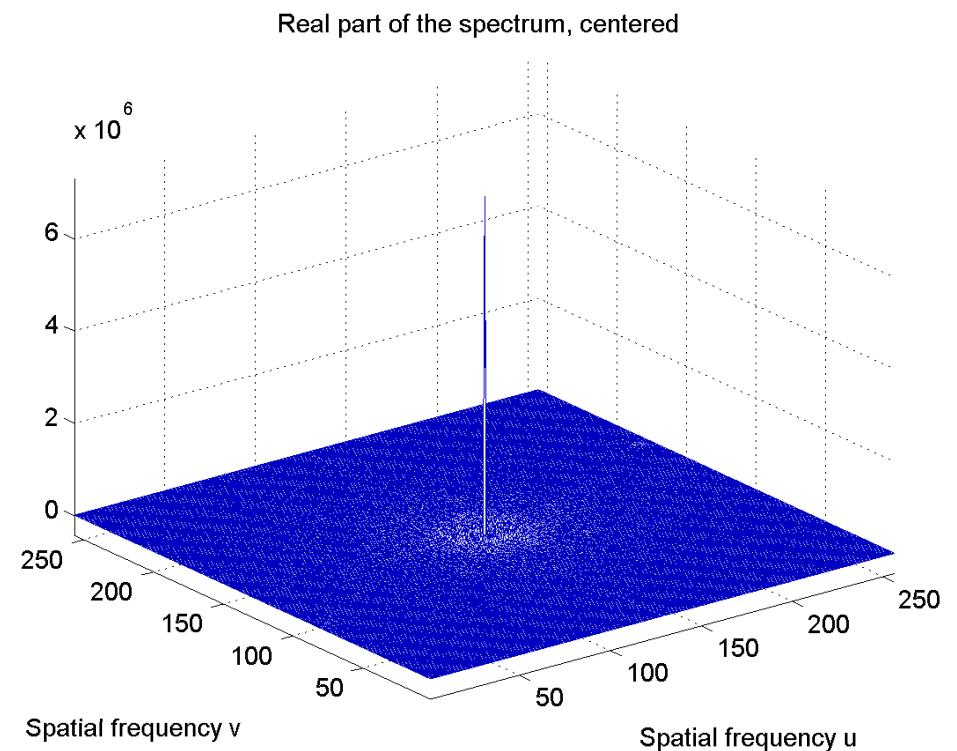
Rice example, input image 265×256



Real part of the centered spectrum, image and mesh

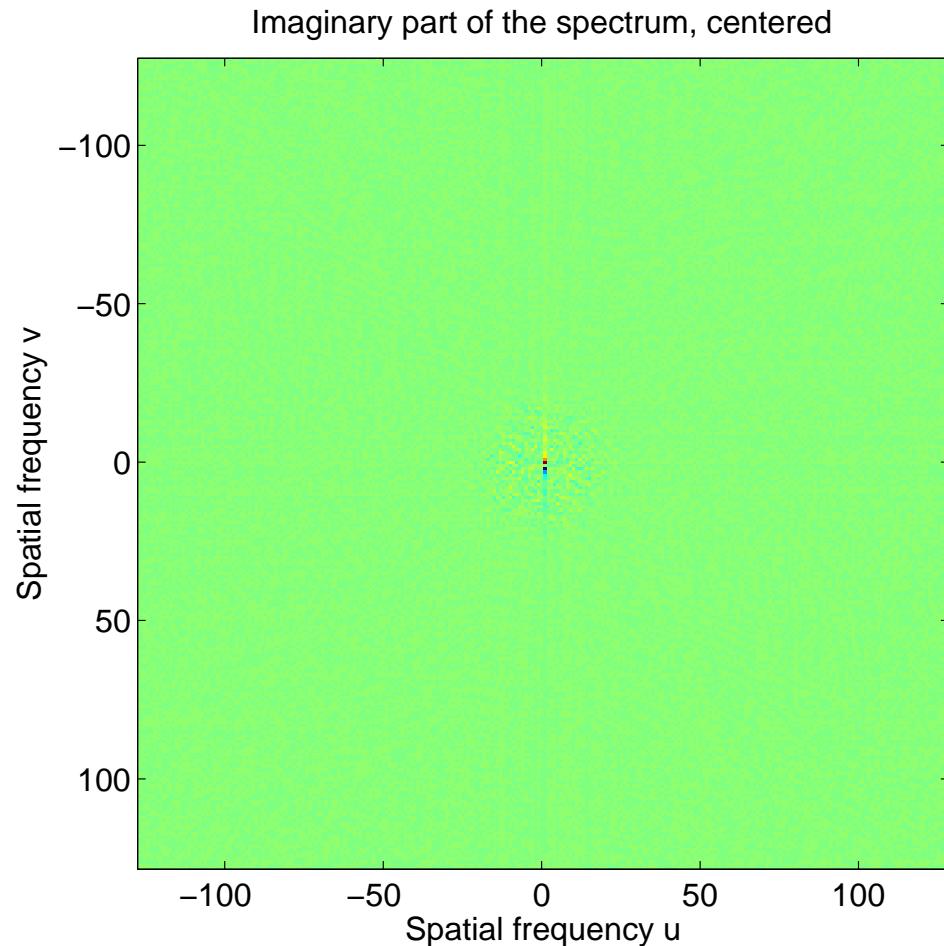


real part, image

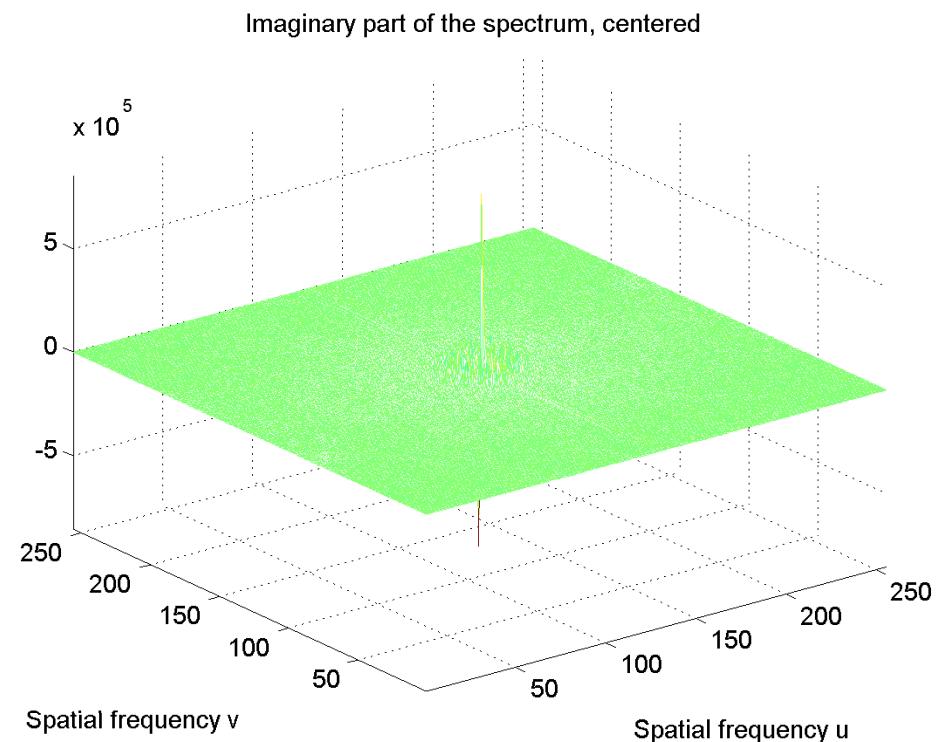


real part, mesh

Imaginary part of the centered spectrum image and mesh

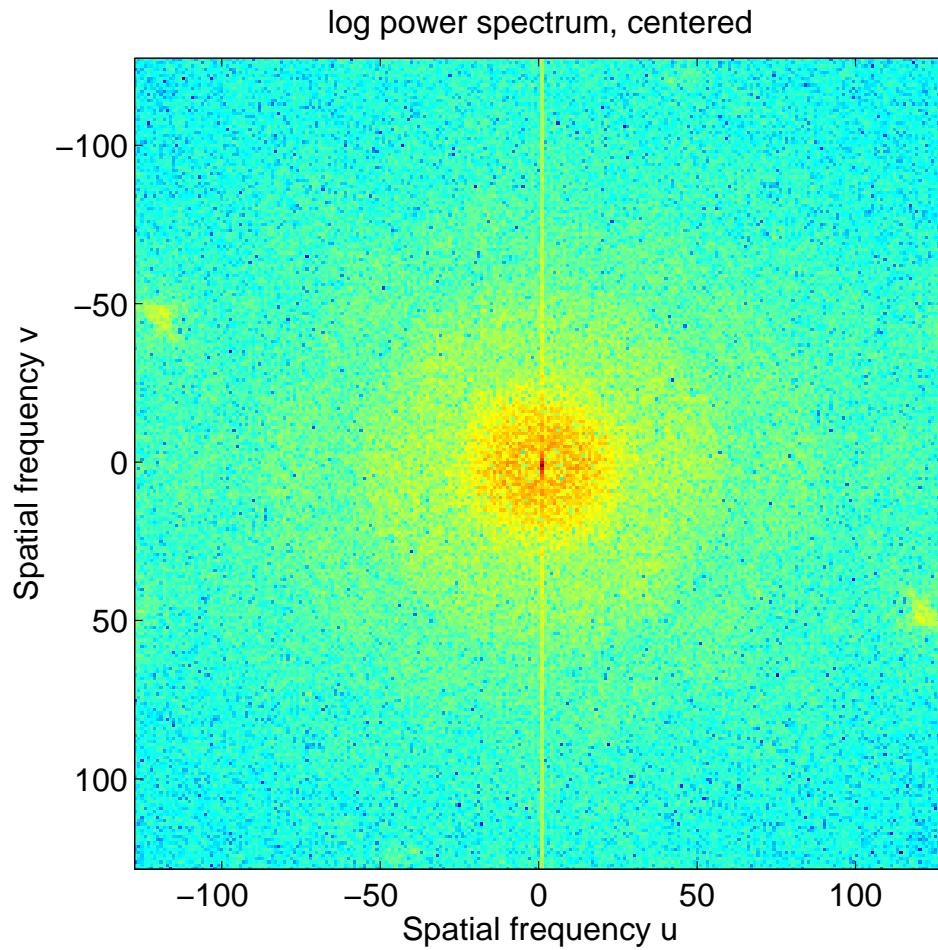


imaginary part, image

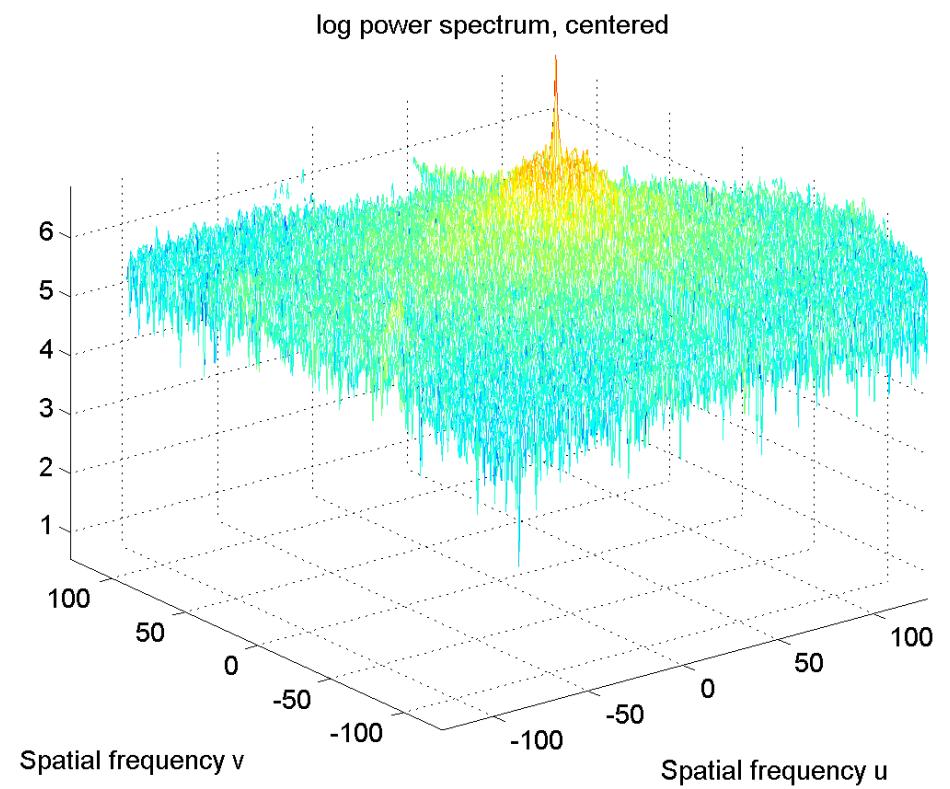


imaginary part, mesh

Log power of the centered spectrum image and mesh

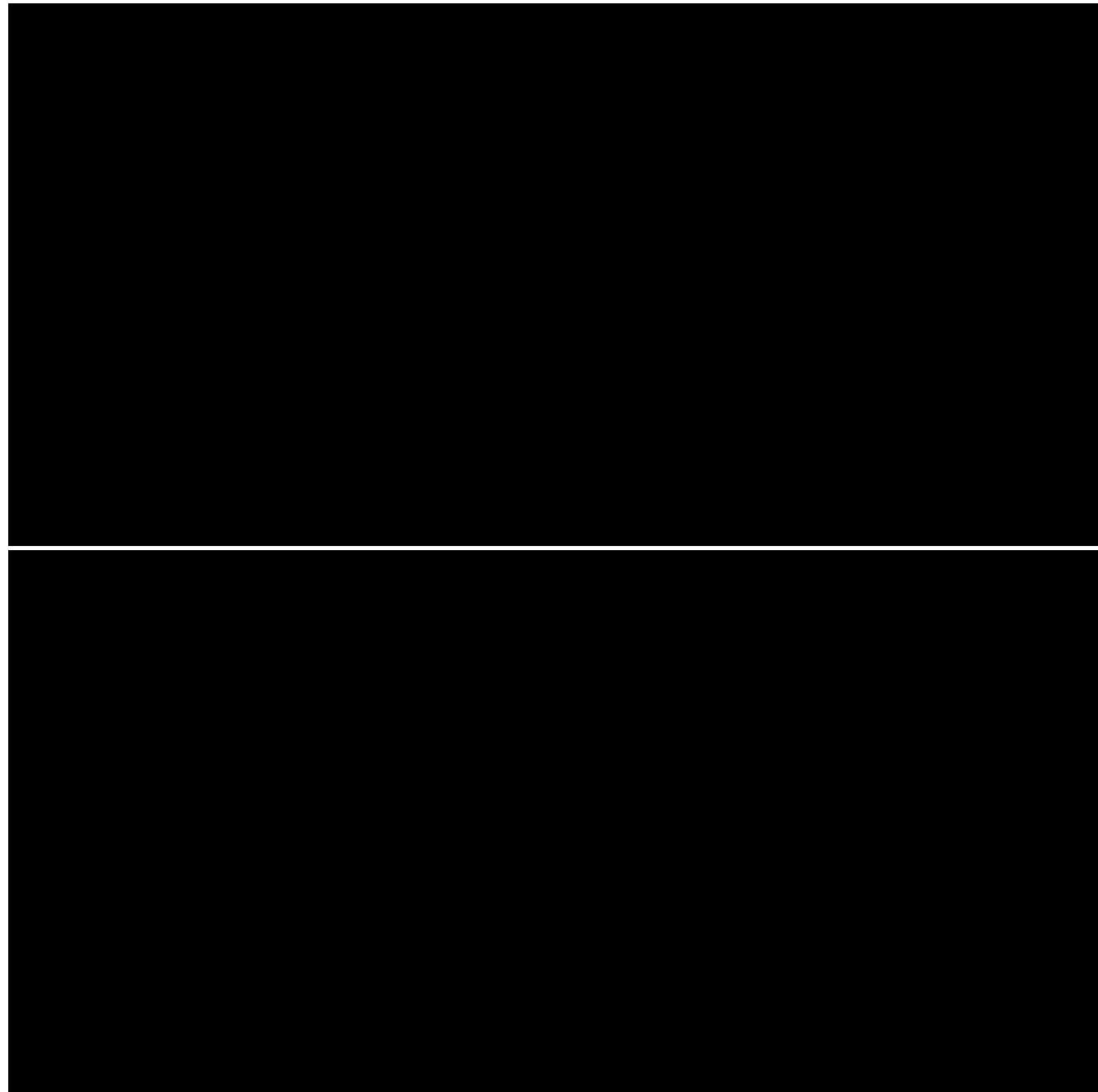


image

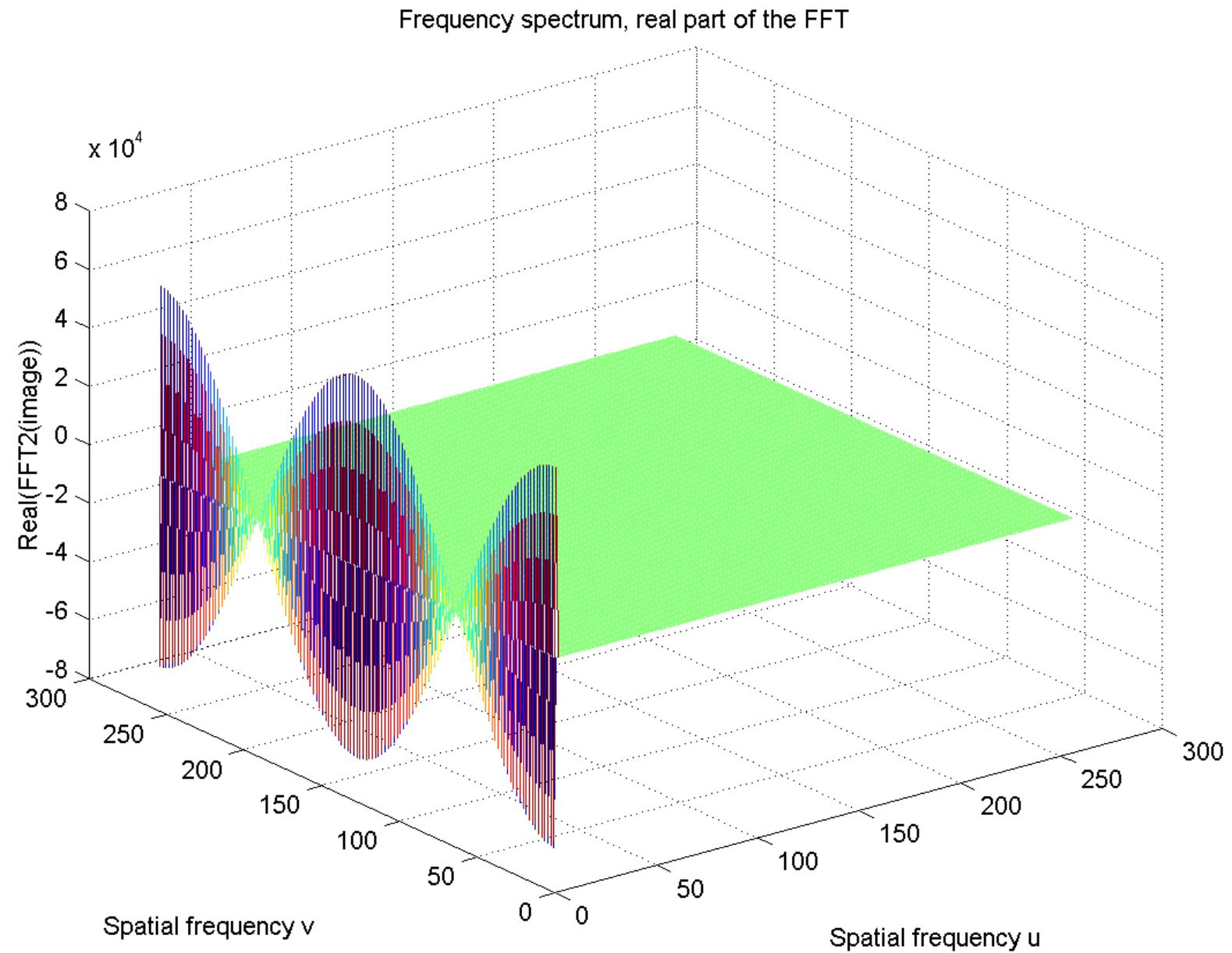


mesh

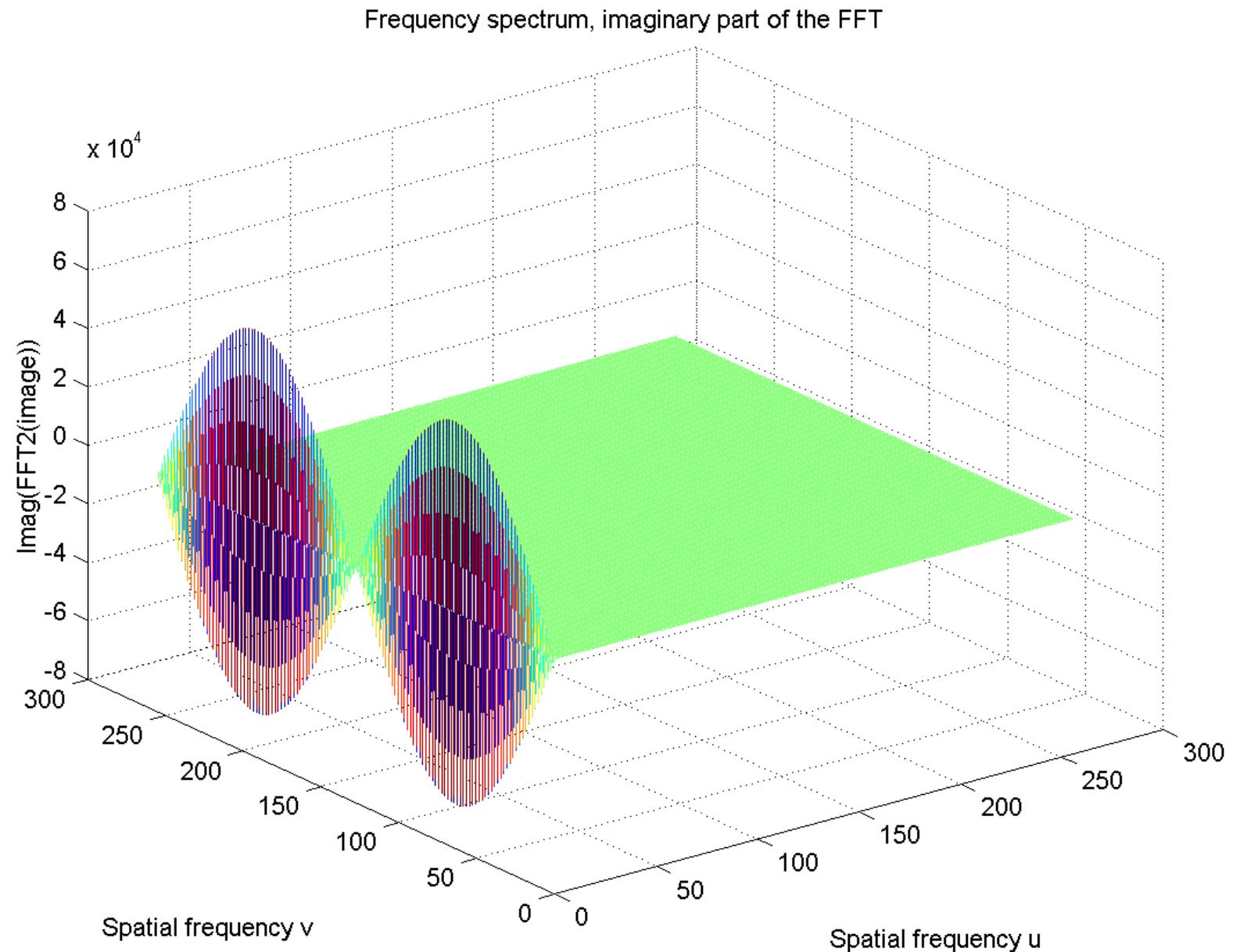
Horizontal line example, input image 265×256



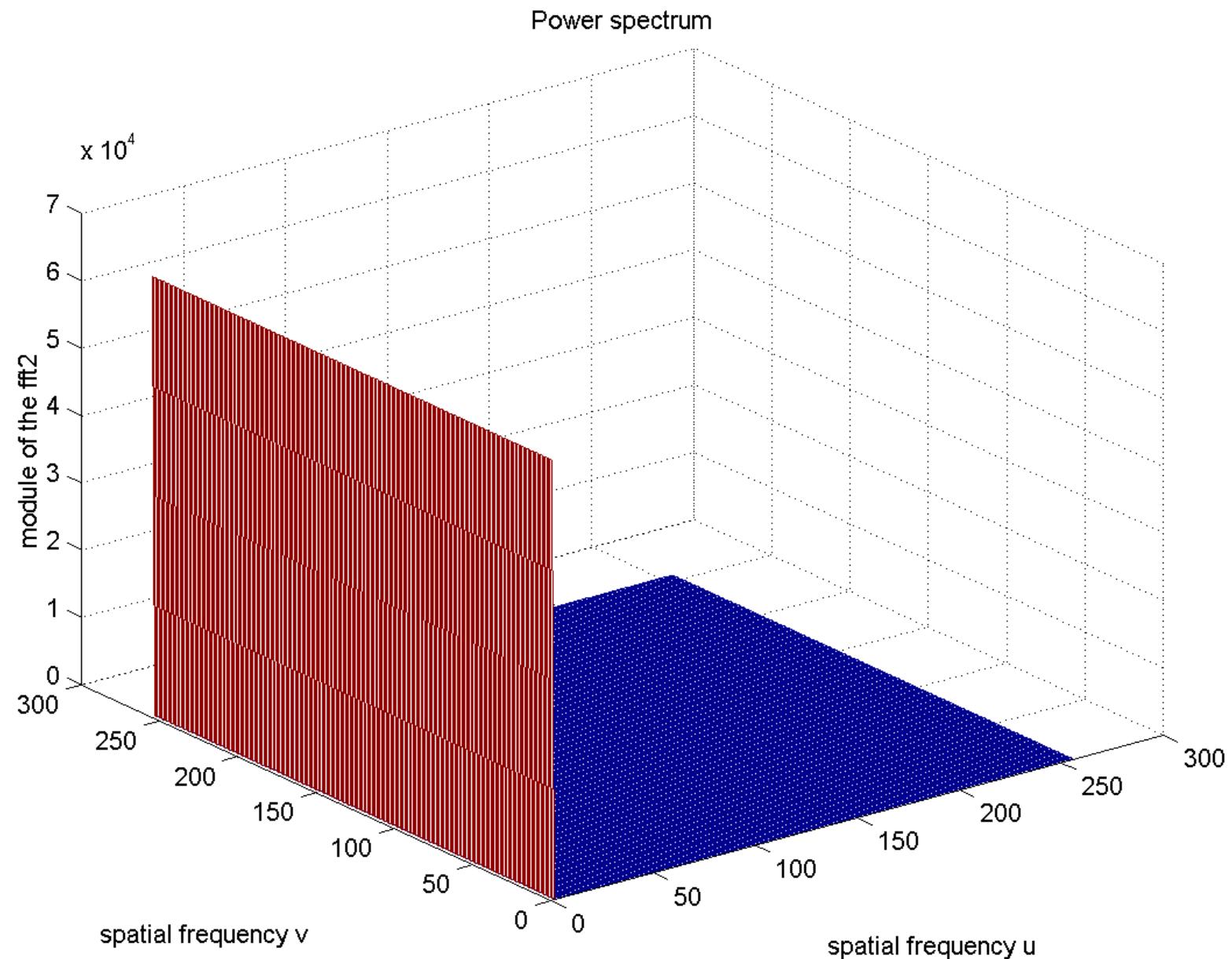
Horizontal line example, real part of the spectrum



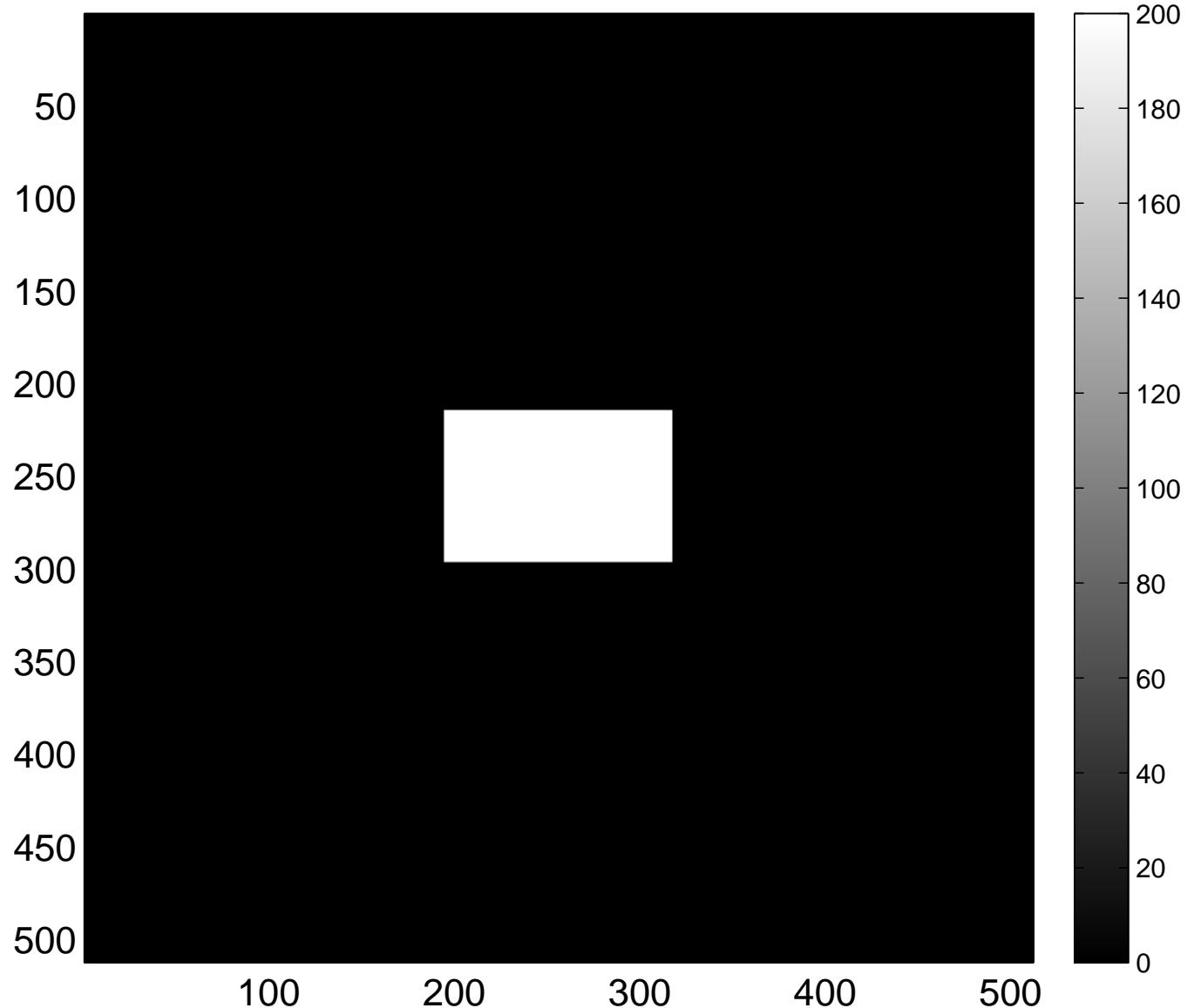
Horizontal line example, imaginary part of the spectrum



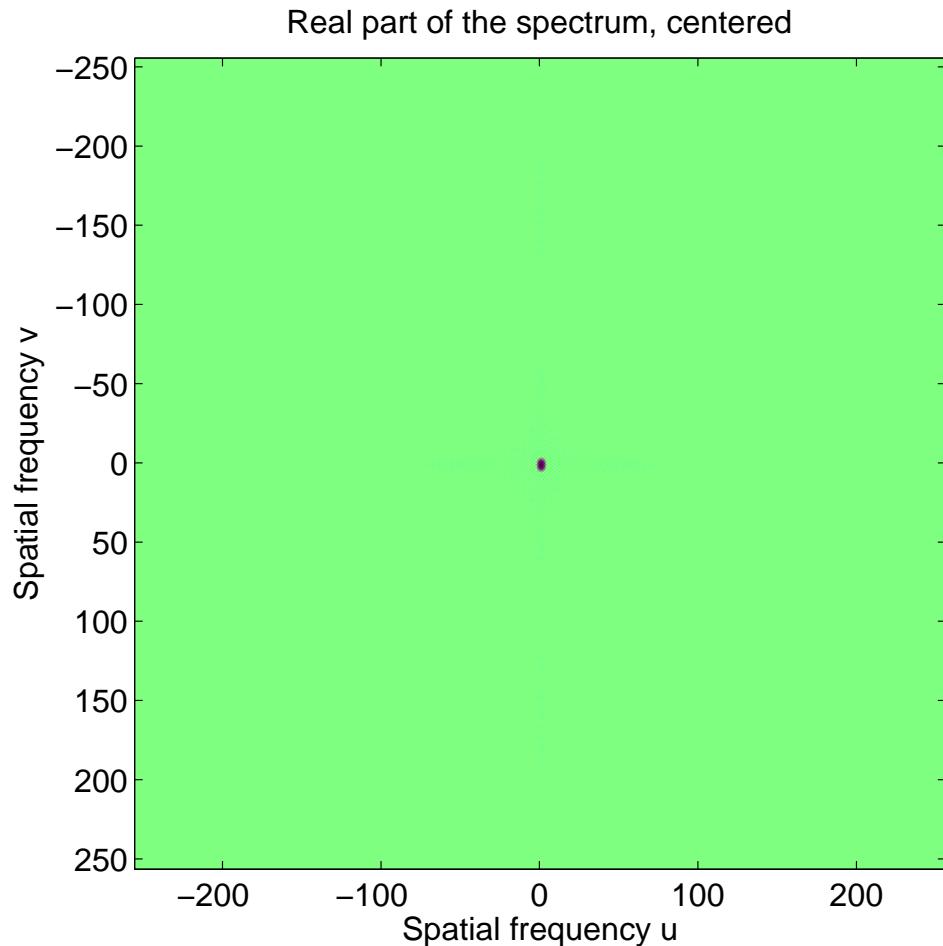
Horizontal line example, power spectrum



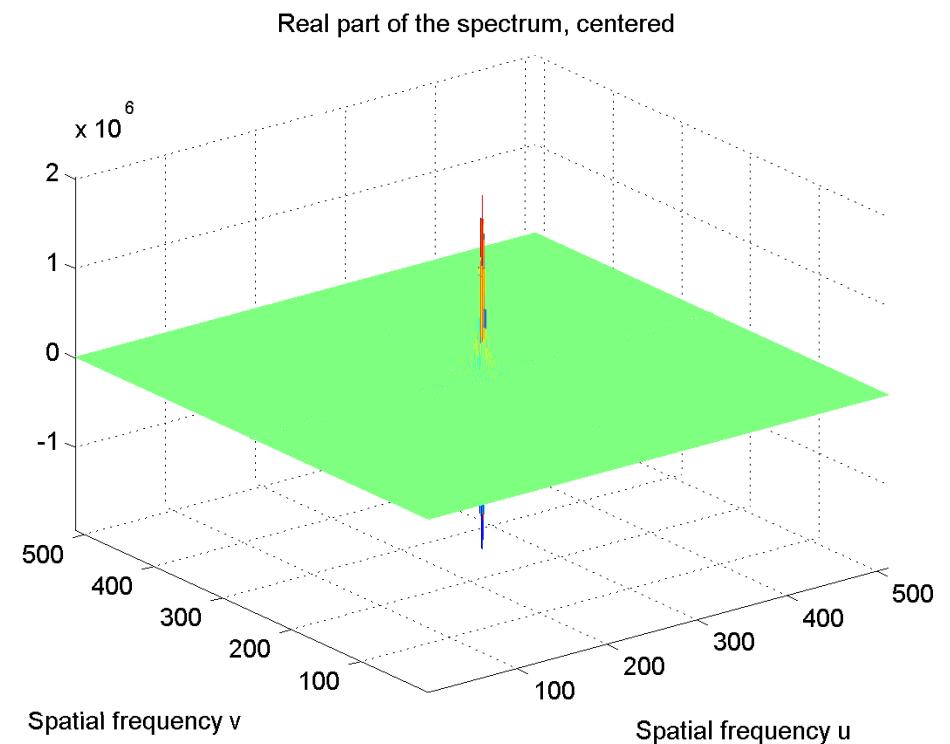
Rectangle example, input image 512×512



Real part of the centered spectrum, image and mesh

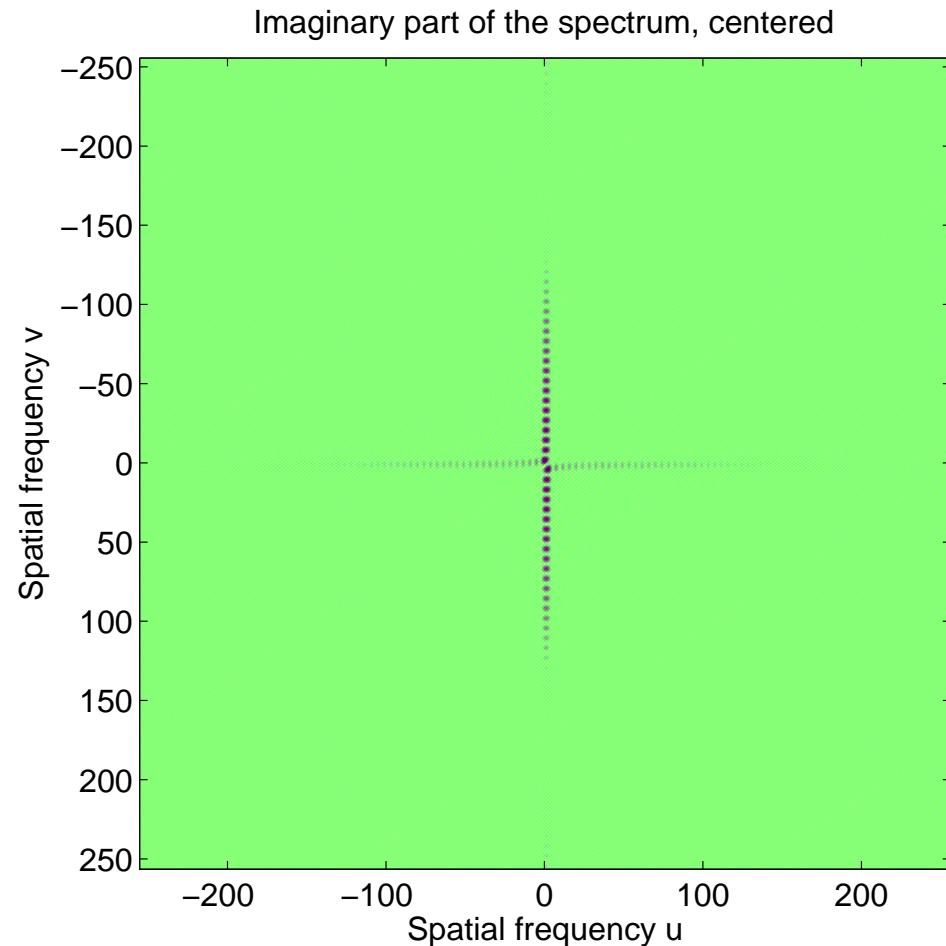


real part, image

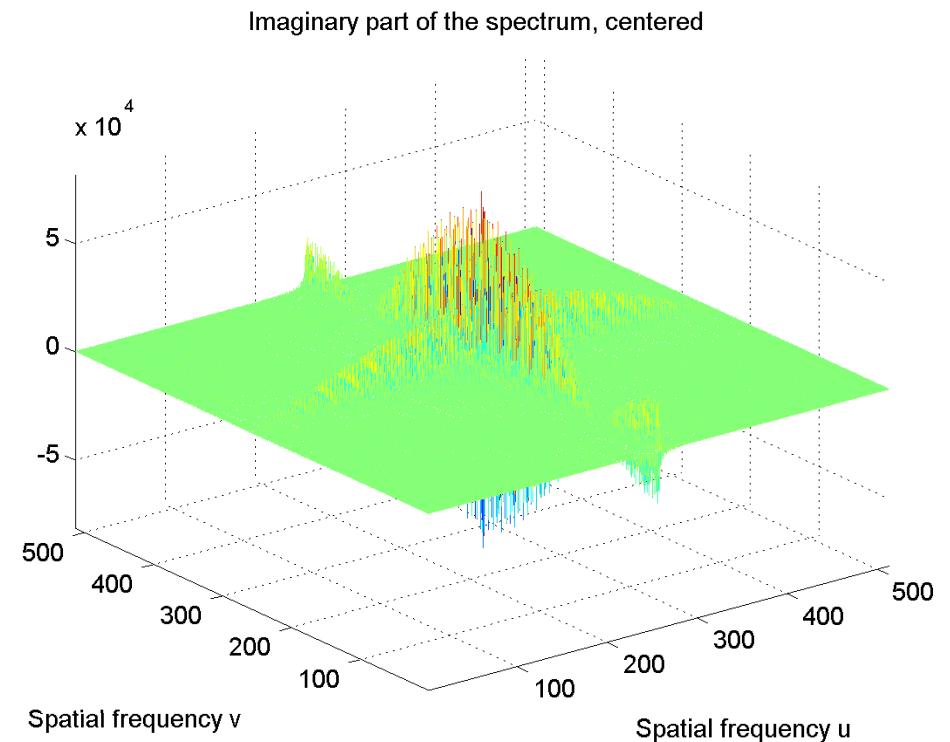


real part, mesh

Imaginary part of the centered spectrum image and mesh

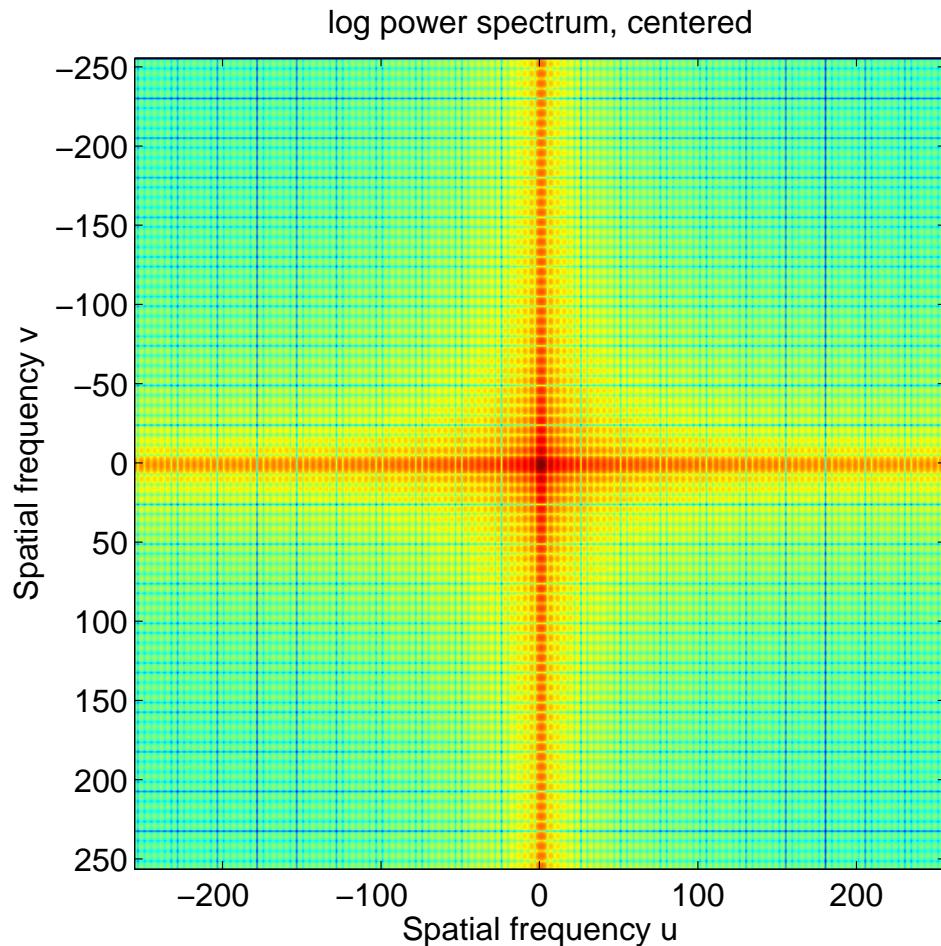


imaginary part, image

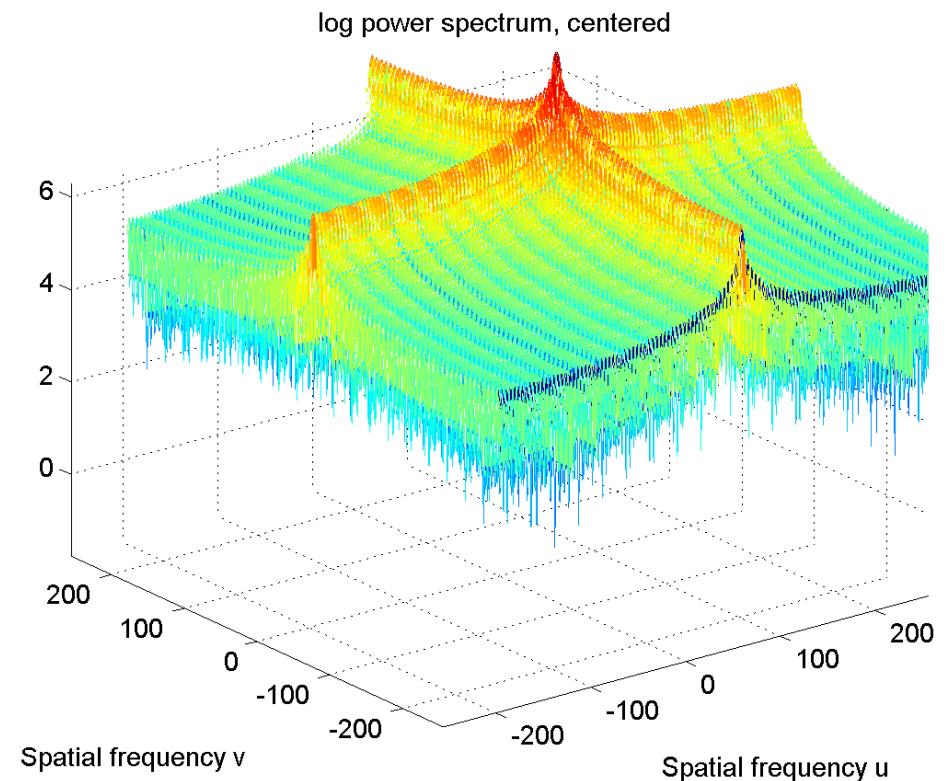


imaginary part, mesh

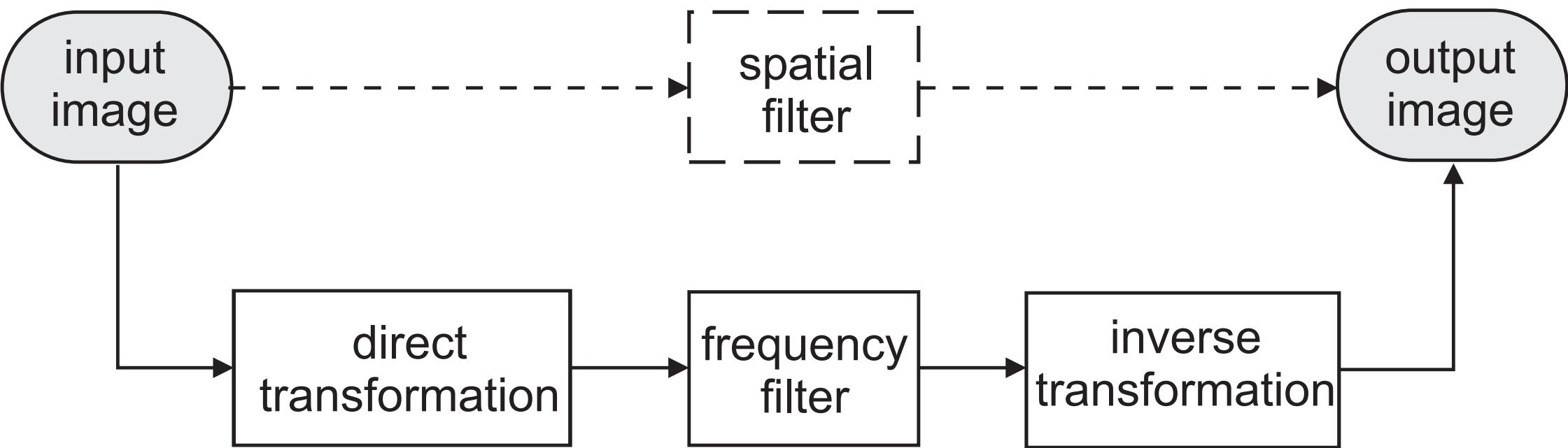
Log power of the centered spectrum image and mesh



image



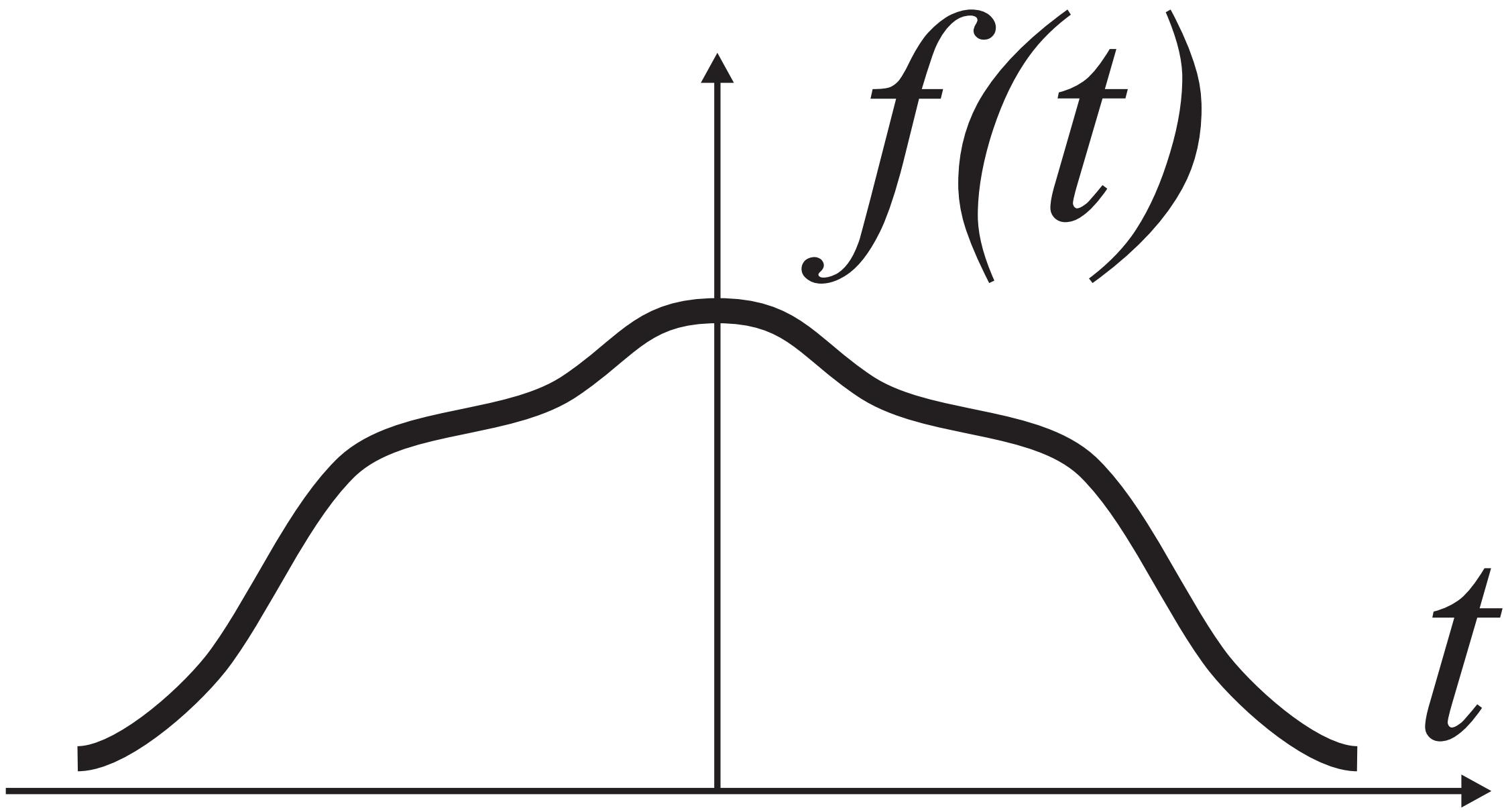
mesh

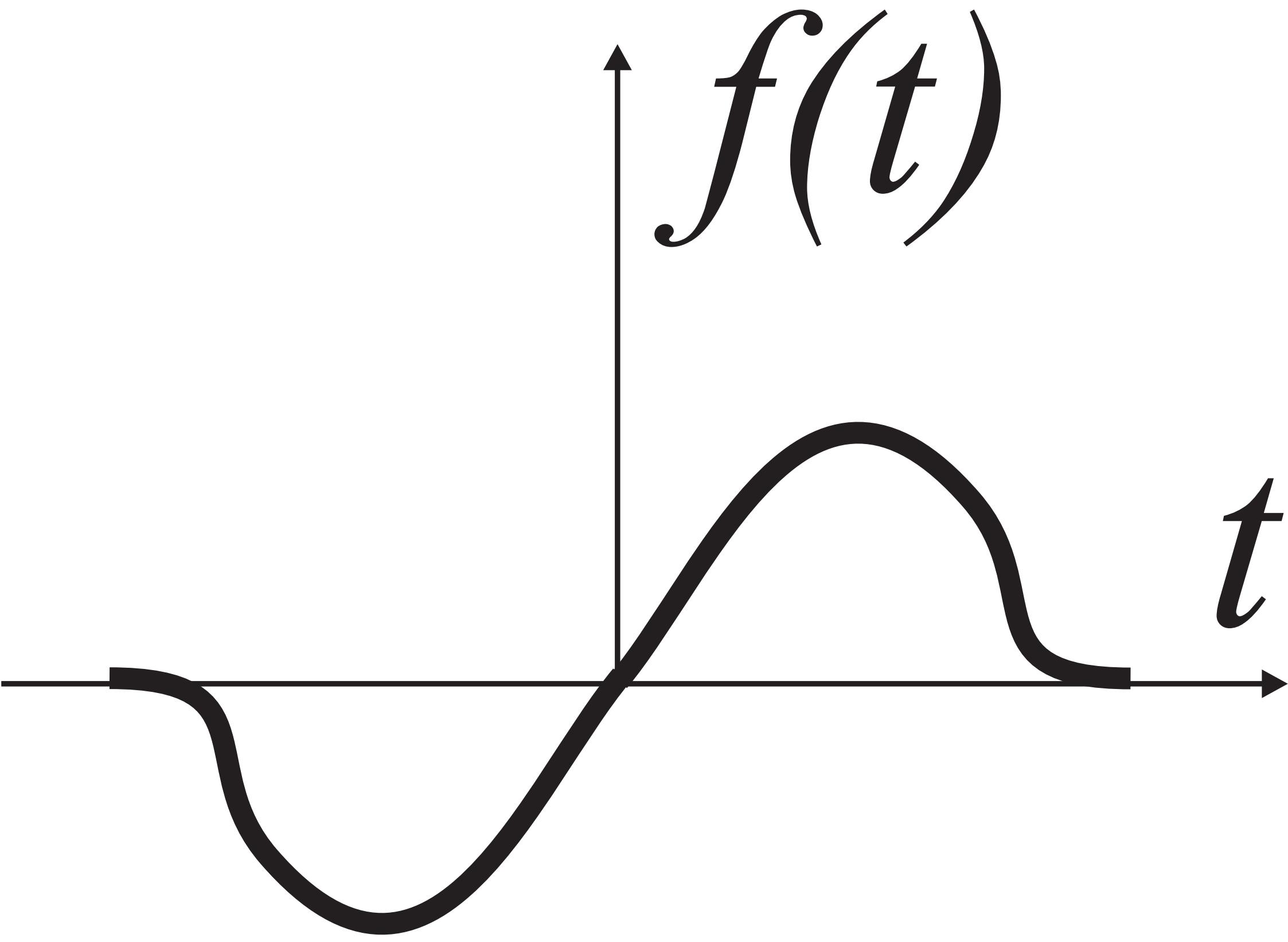


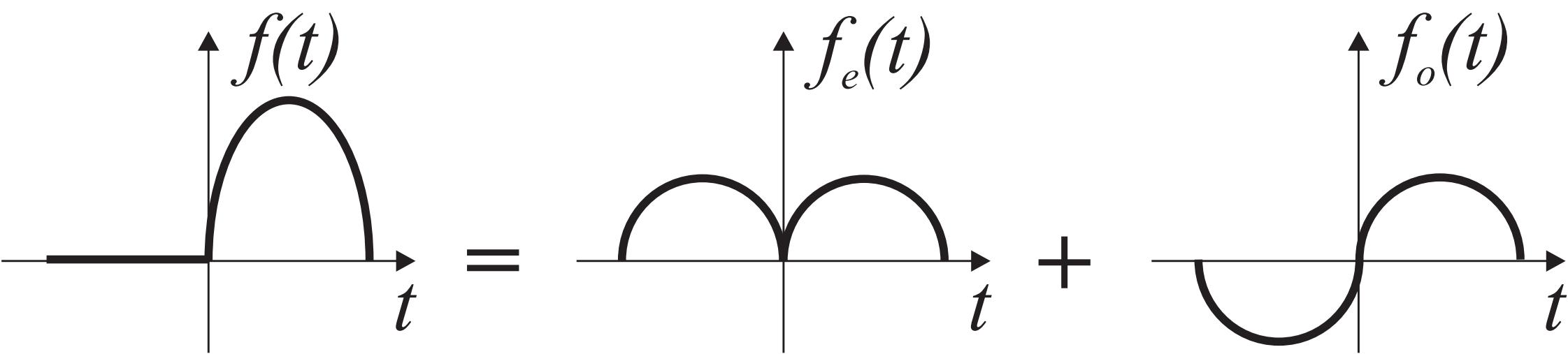


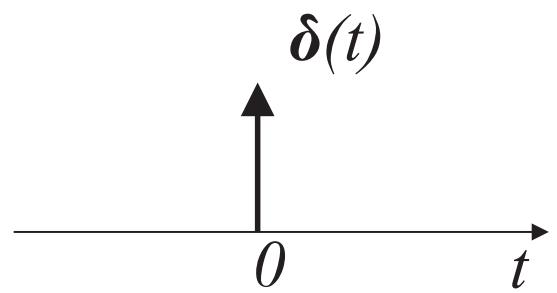
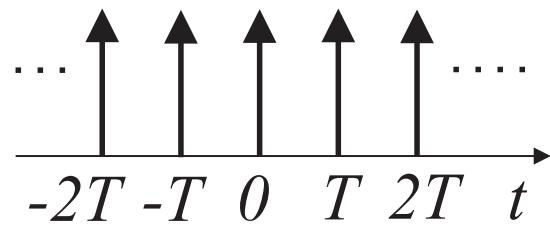
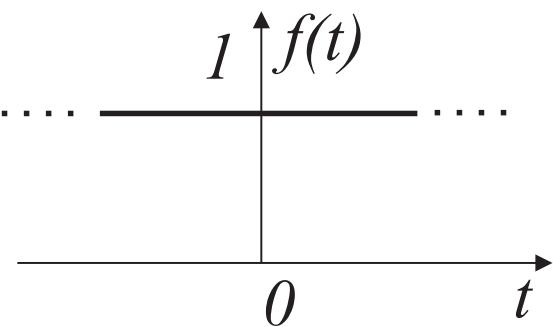
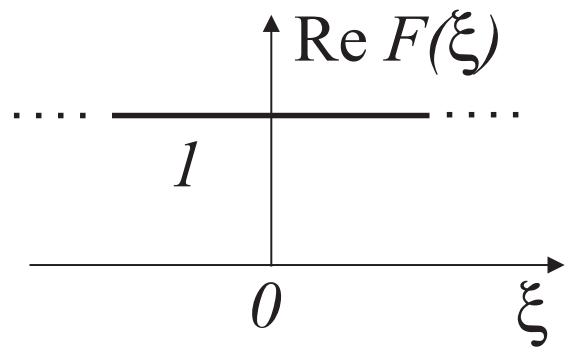
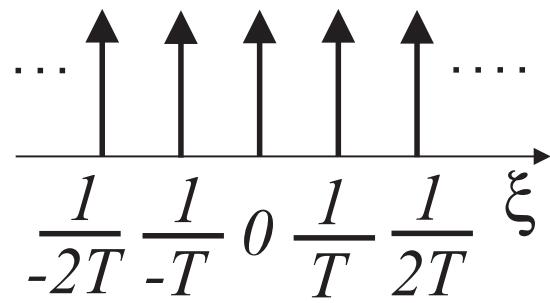
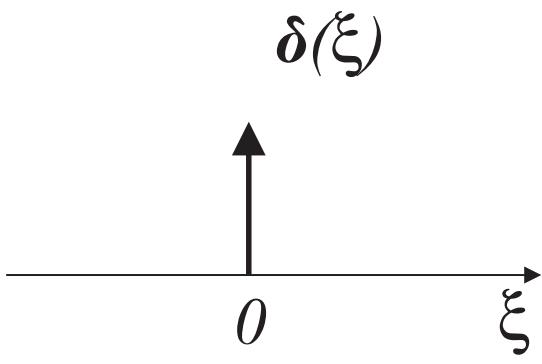
J. Bodly Del.

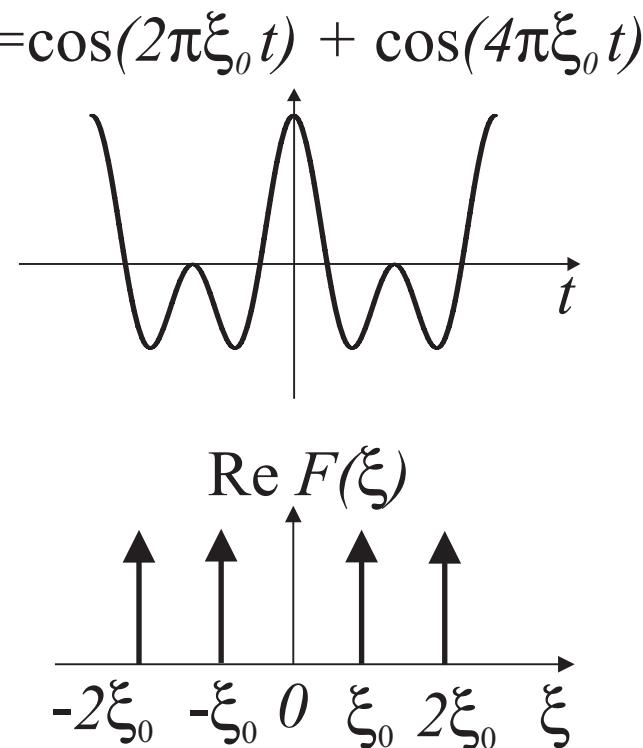
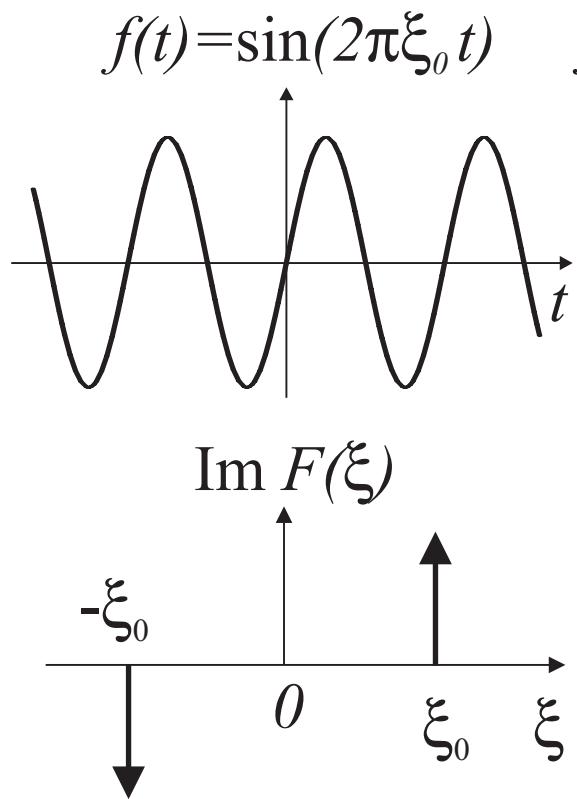
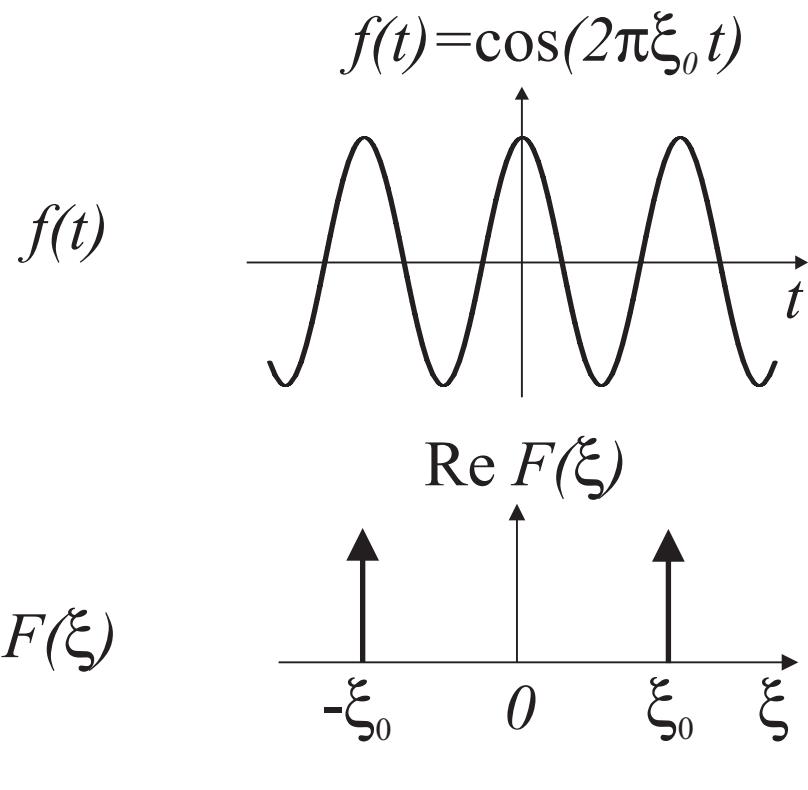
Geille Sculp.

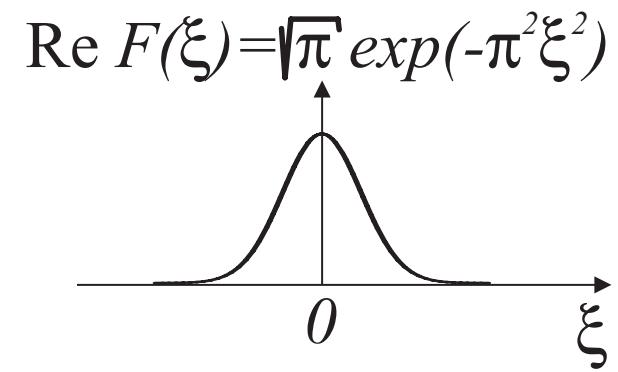
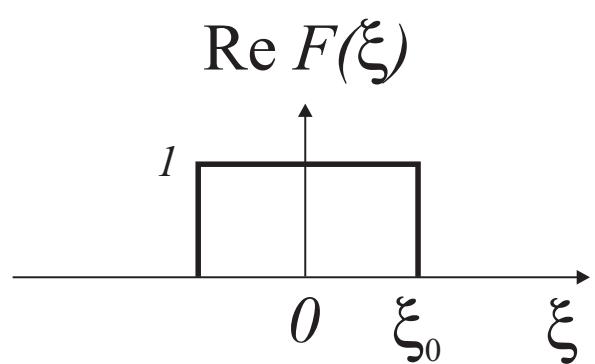
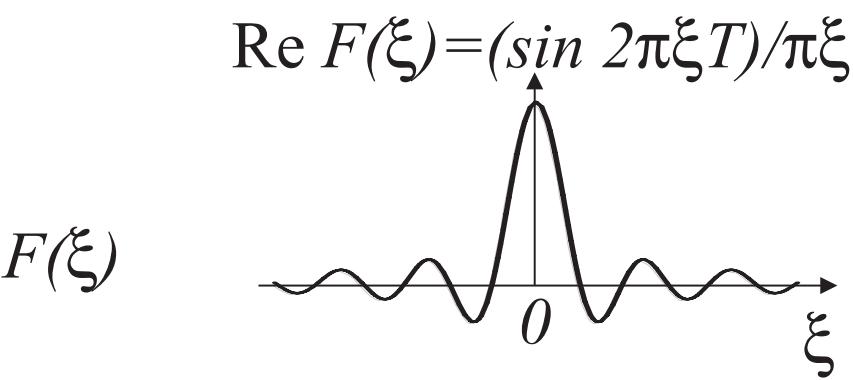
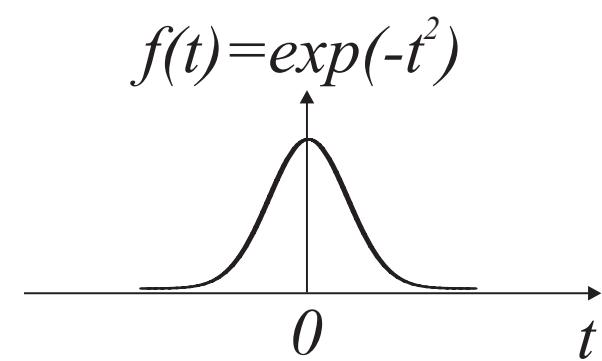
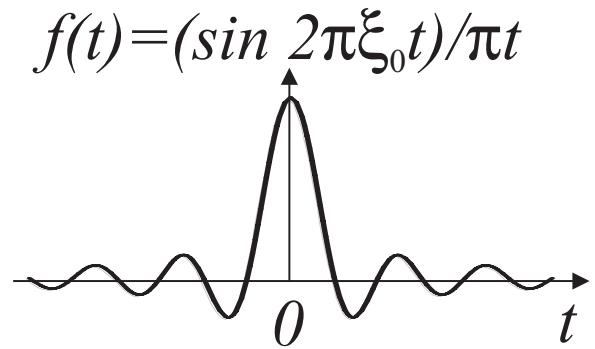
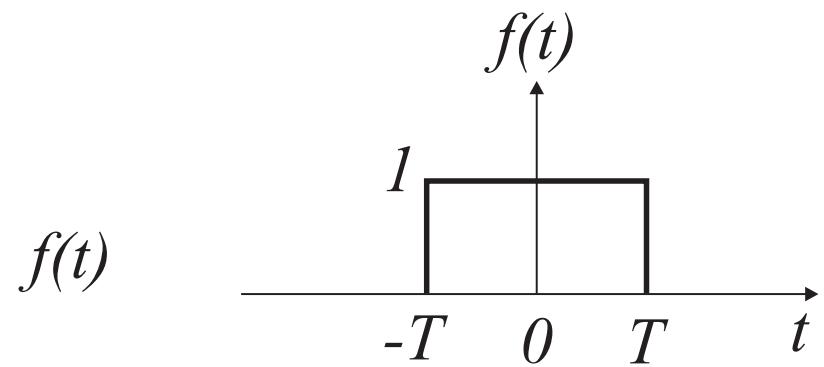


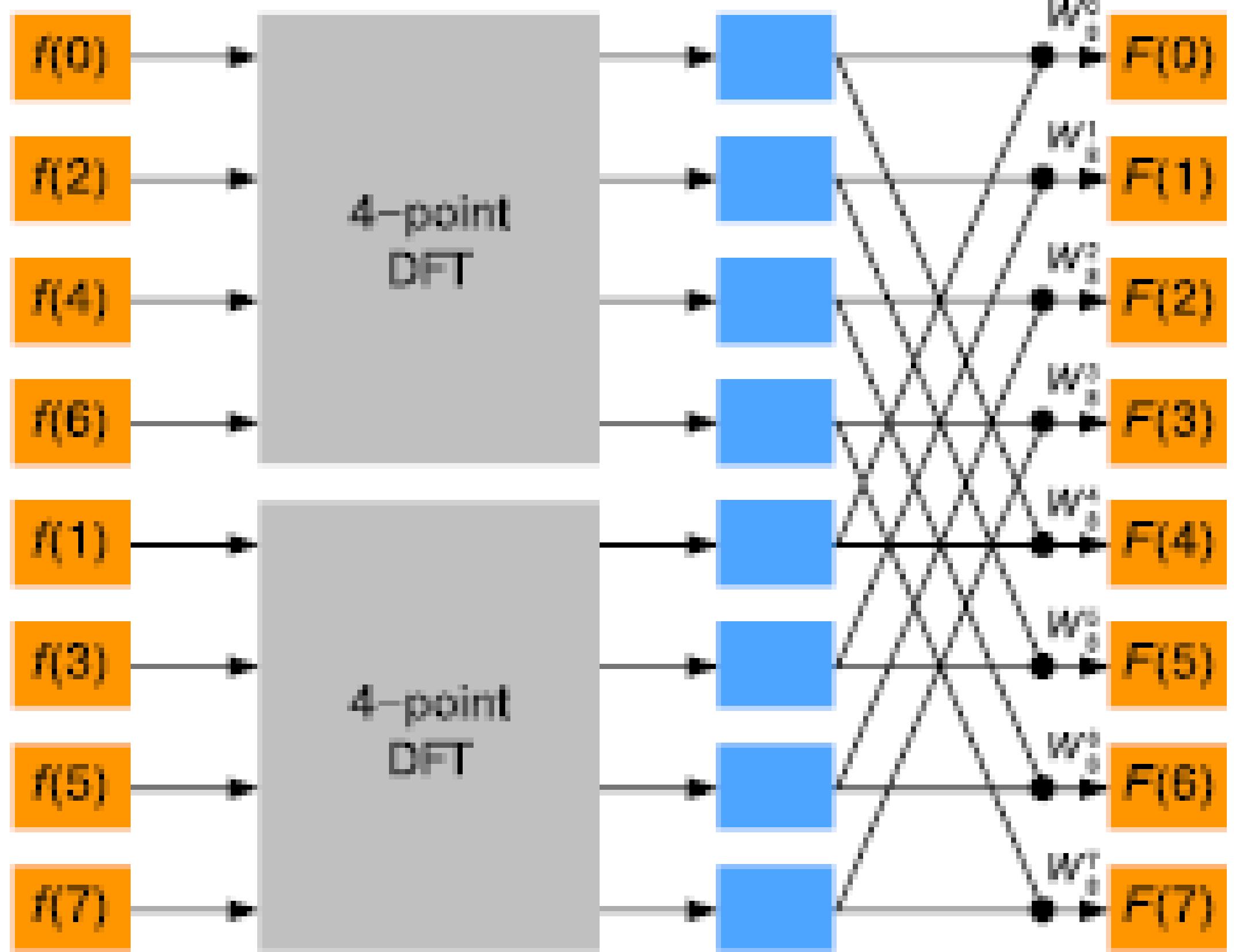


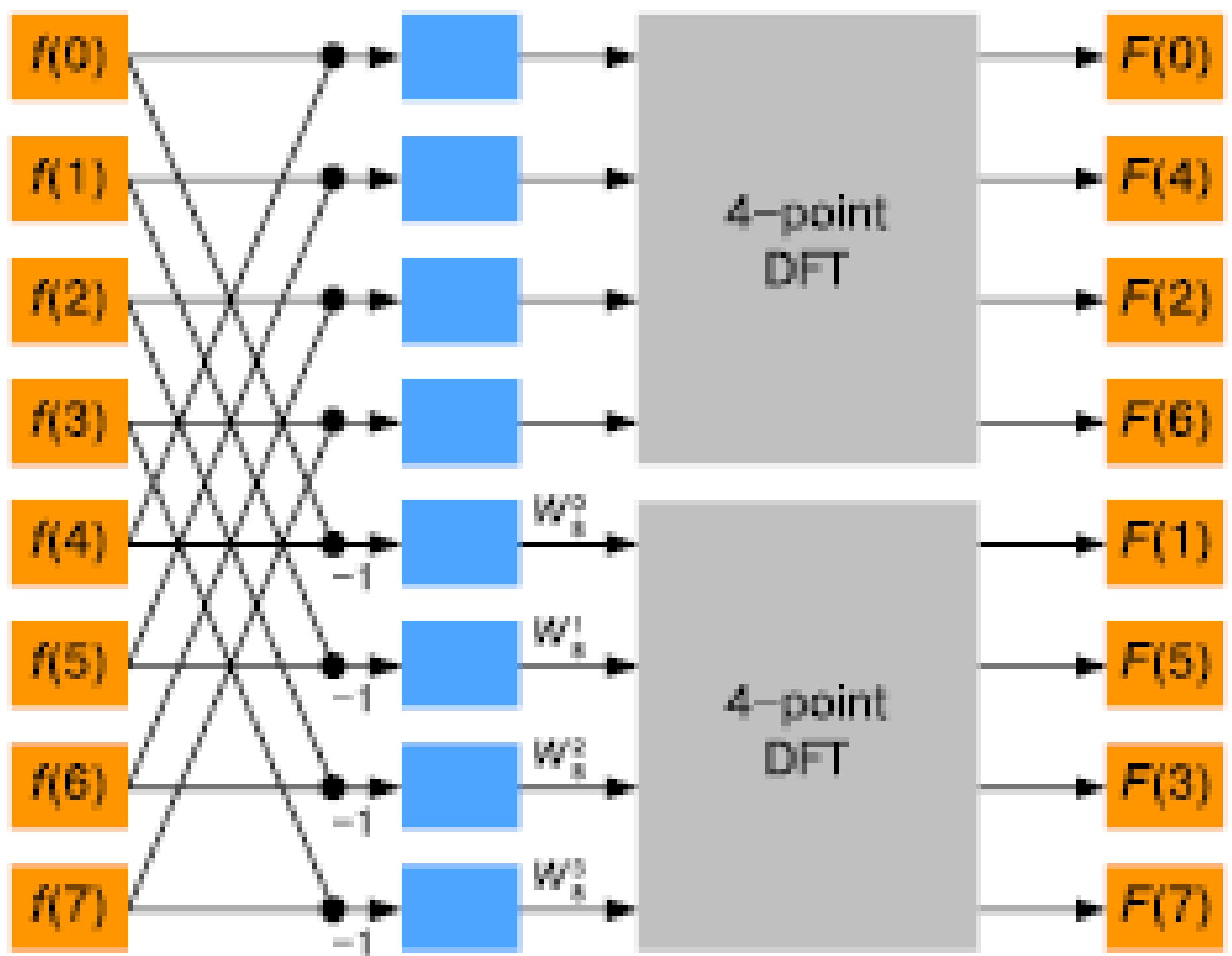


$f(t)$  $f(t)$  $F(\xi)$  $\delta(\xi)$ 









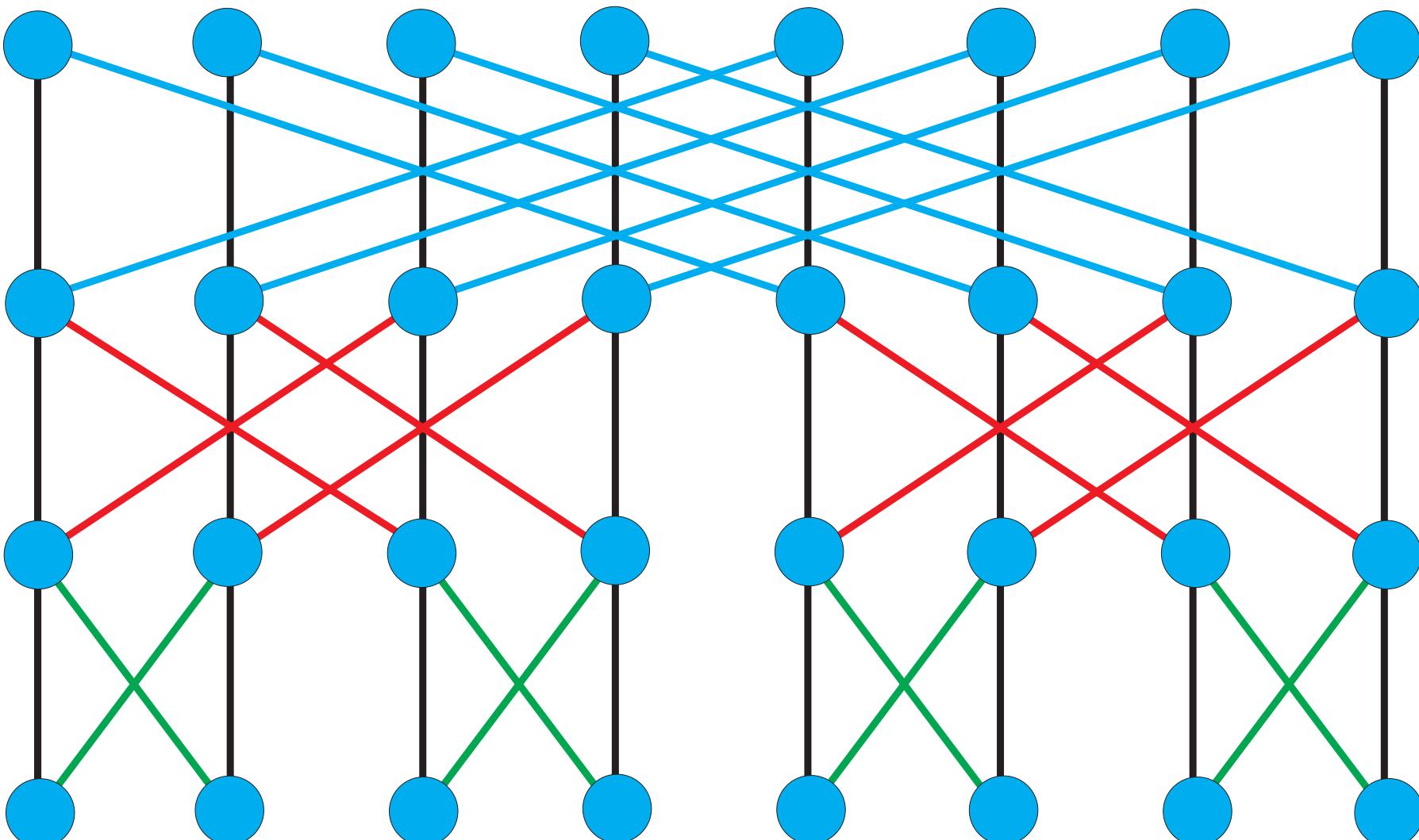
Iteration

1

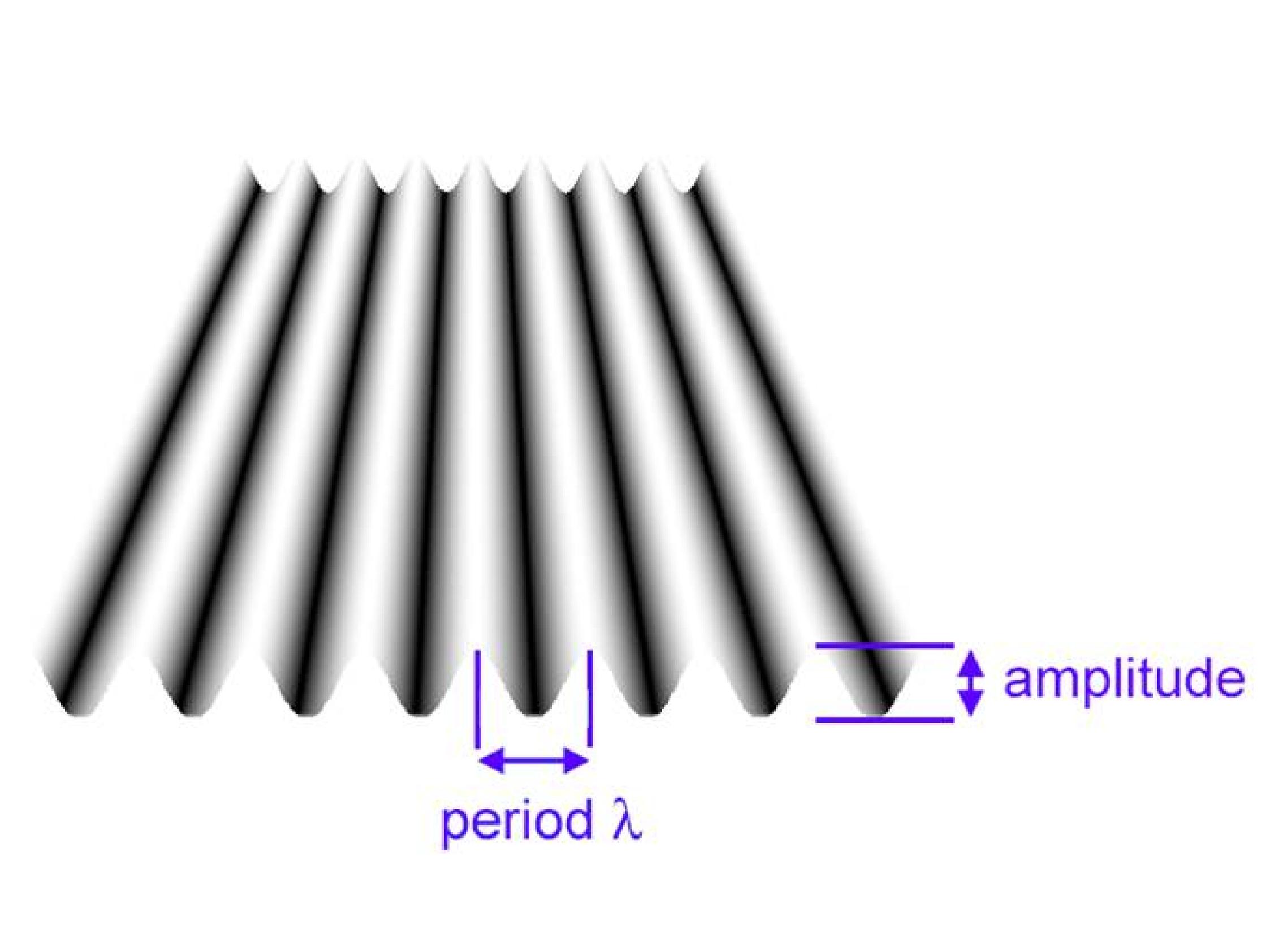
2

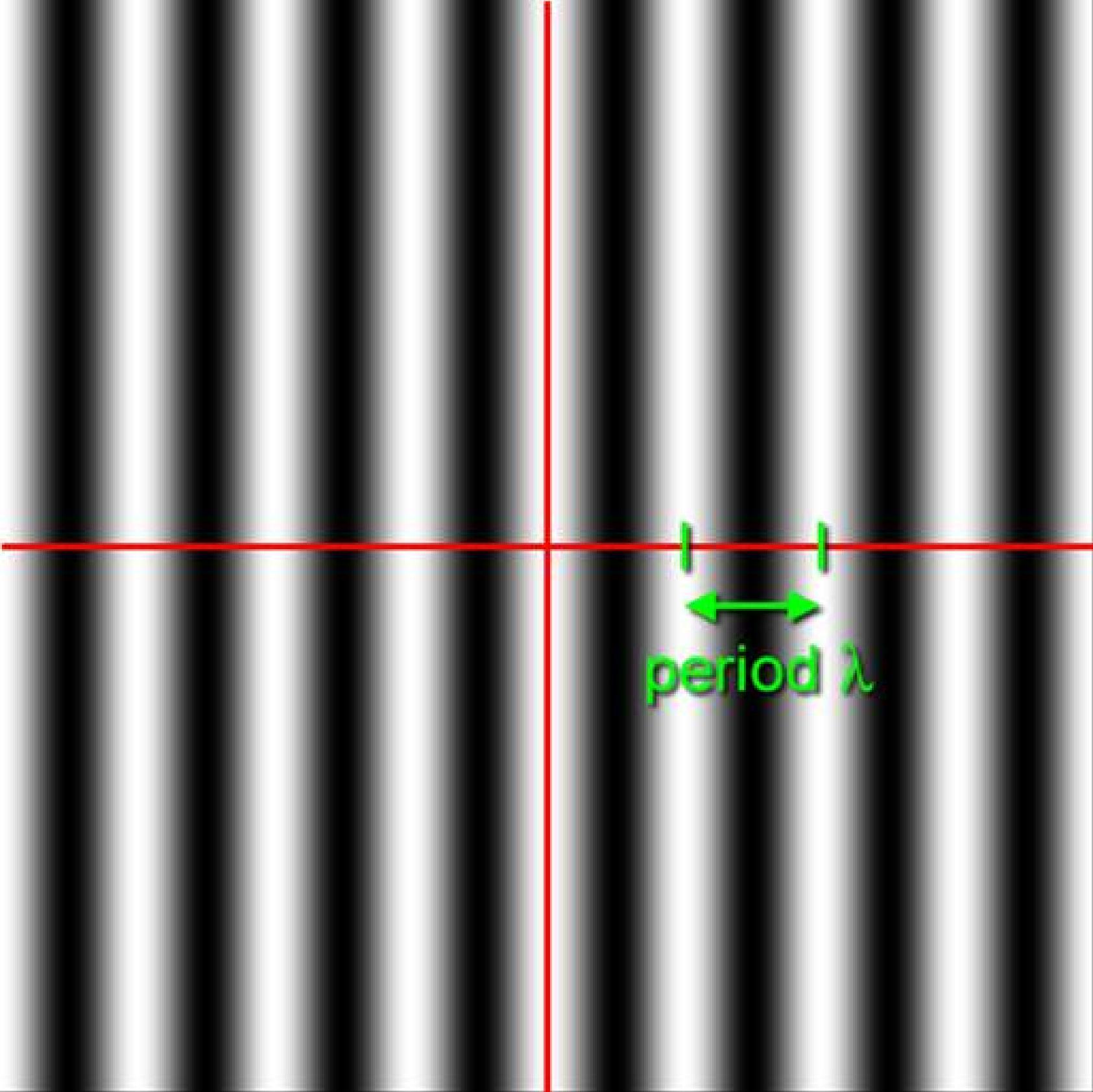
3

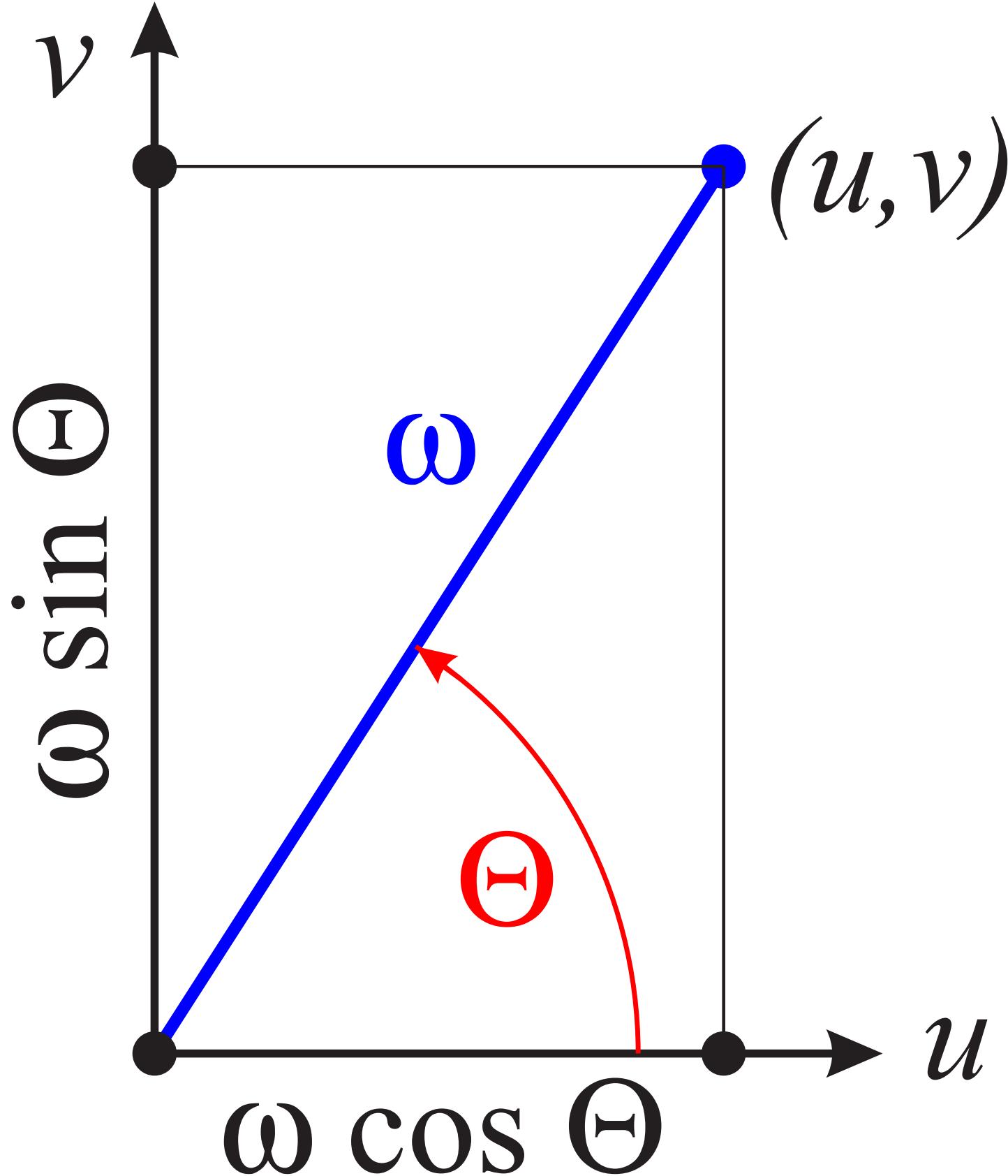
$f_0 \quad f_1 \quad f_2 \quad f_3 \quad f_4 \quad f_5 \quad f_6 \quad f_7$

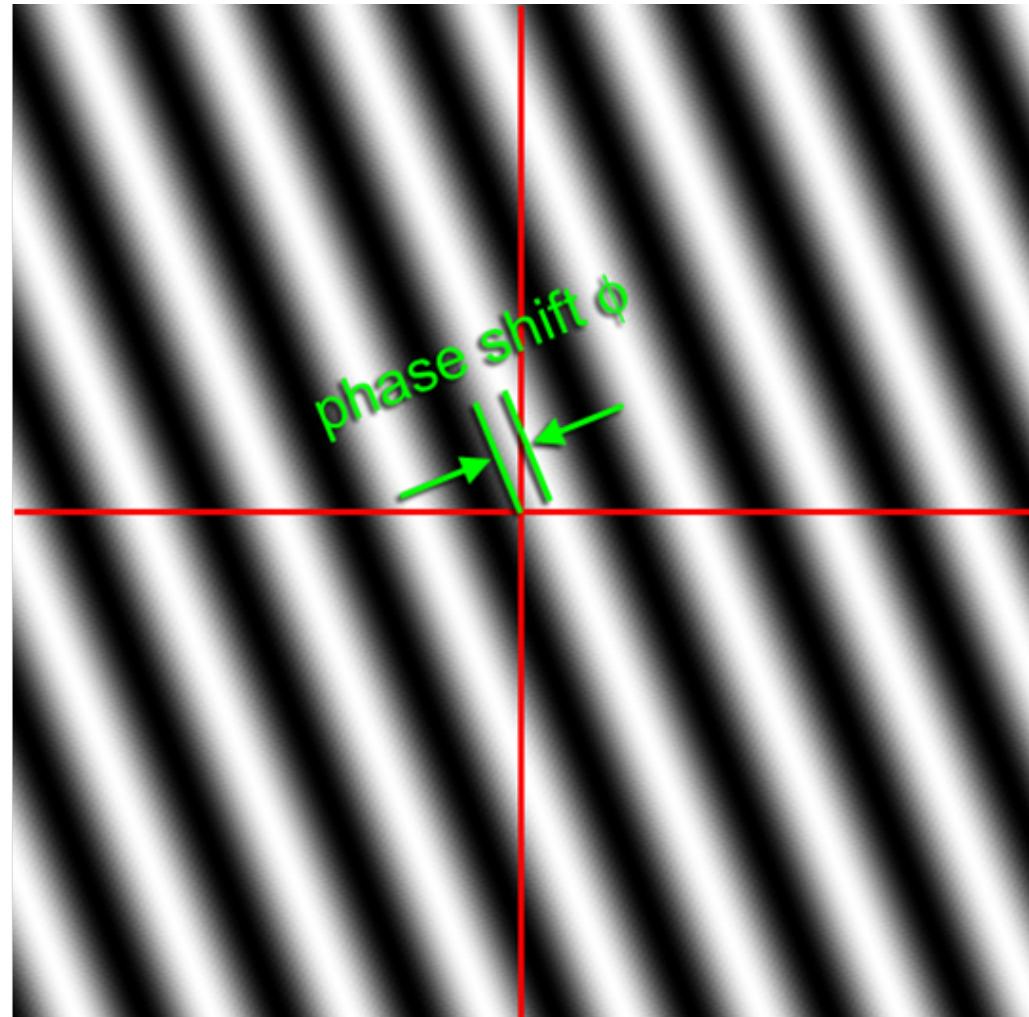
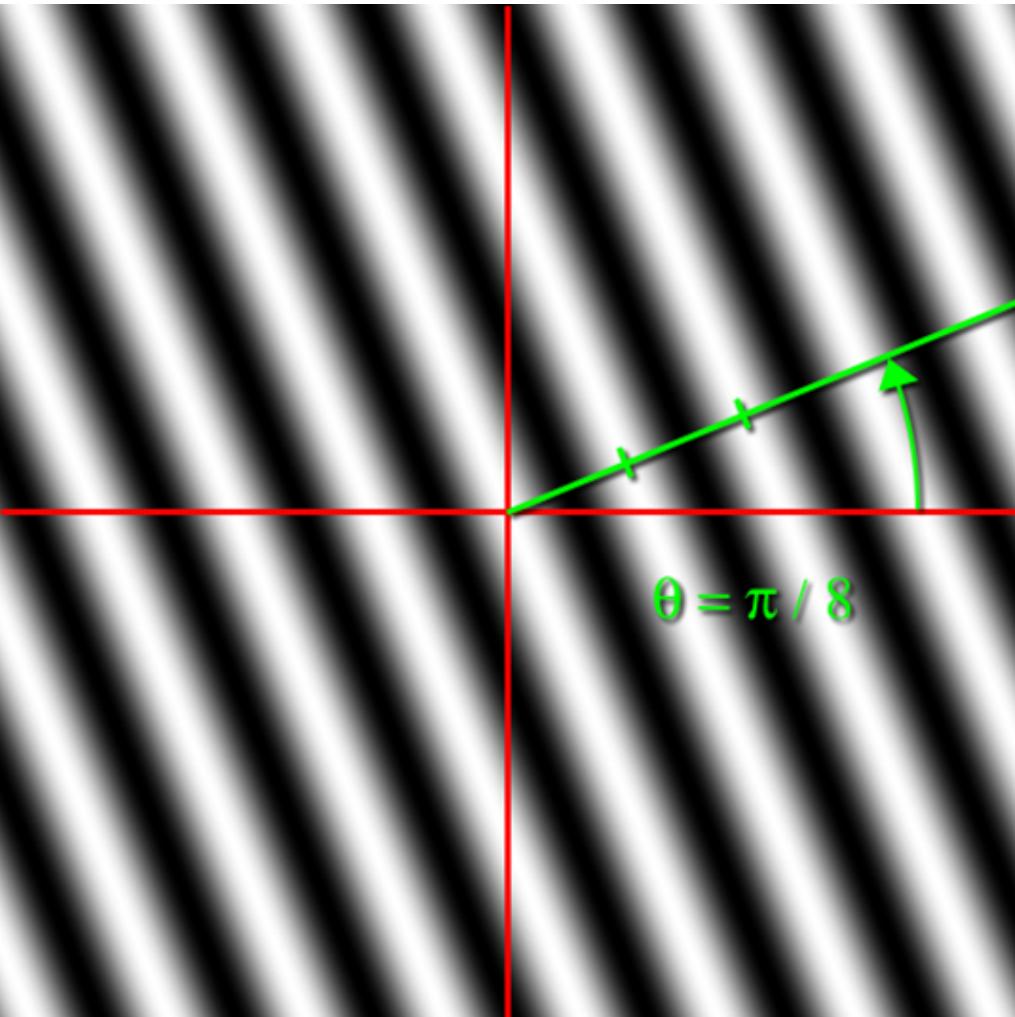


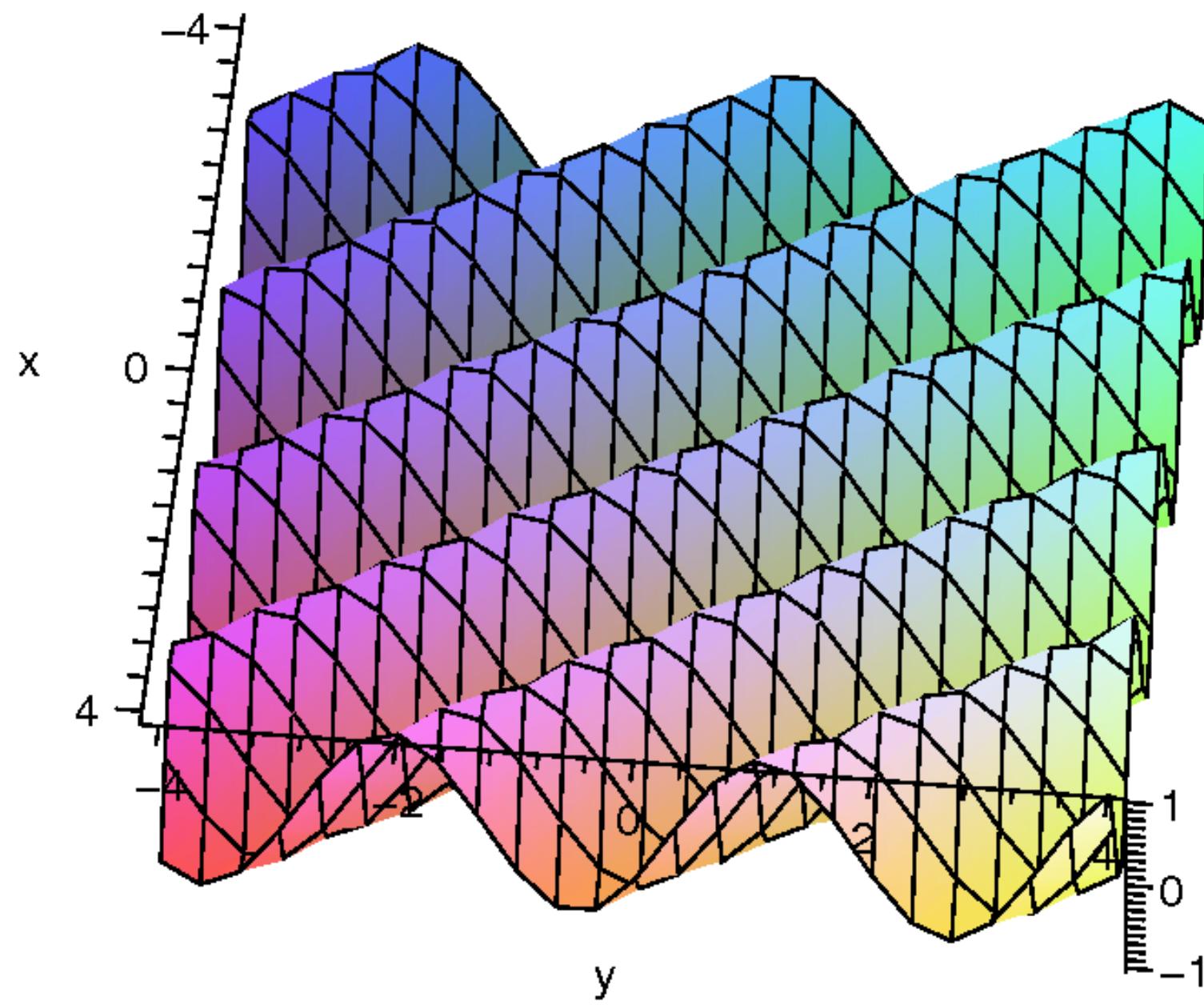
$F_0 \quad F_1 \quad F_2 \quad F_3 \quad F_4 \quad F_5 \quad F_6 \quad F_7$

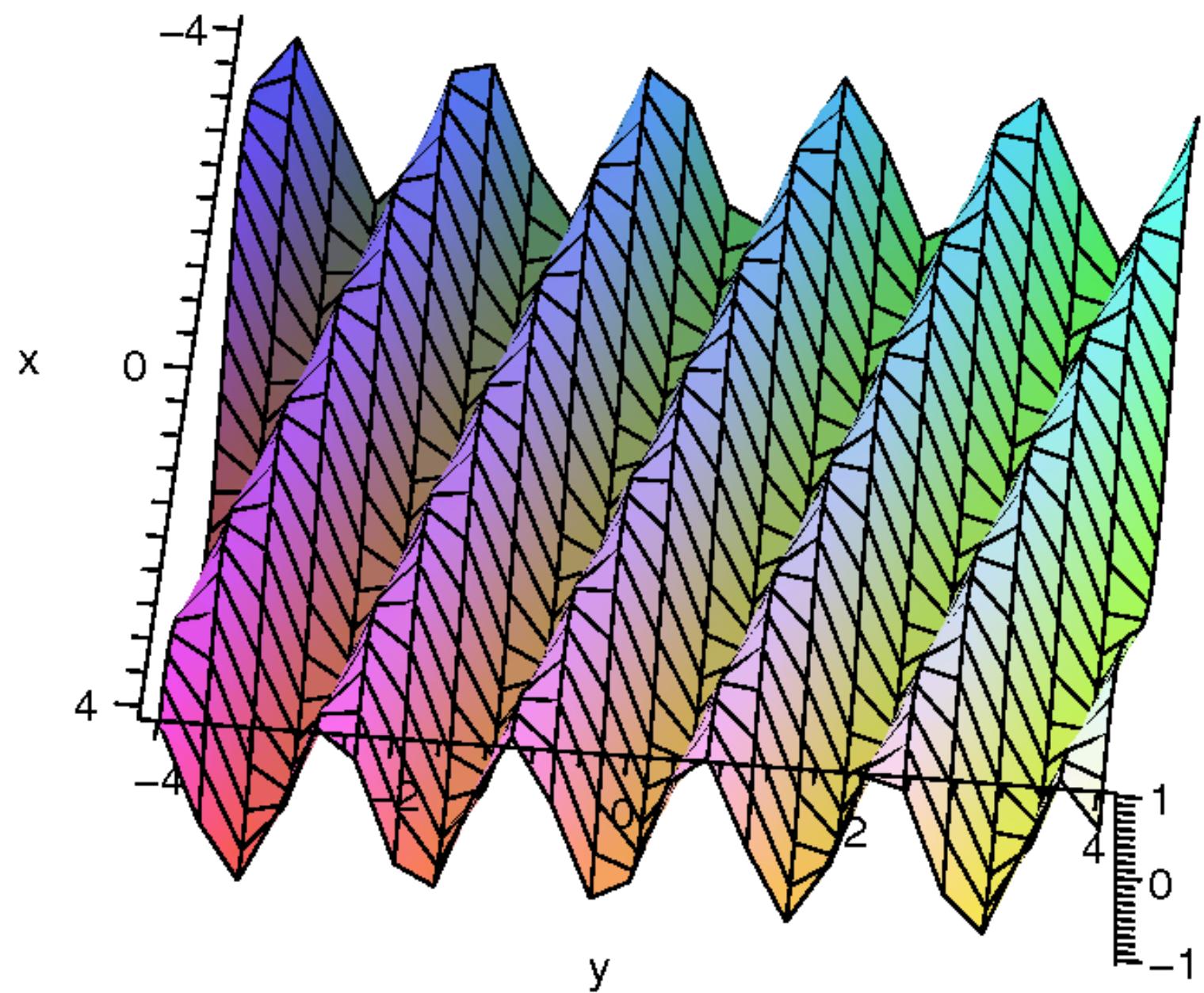


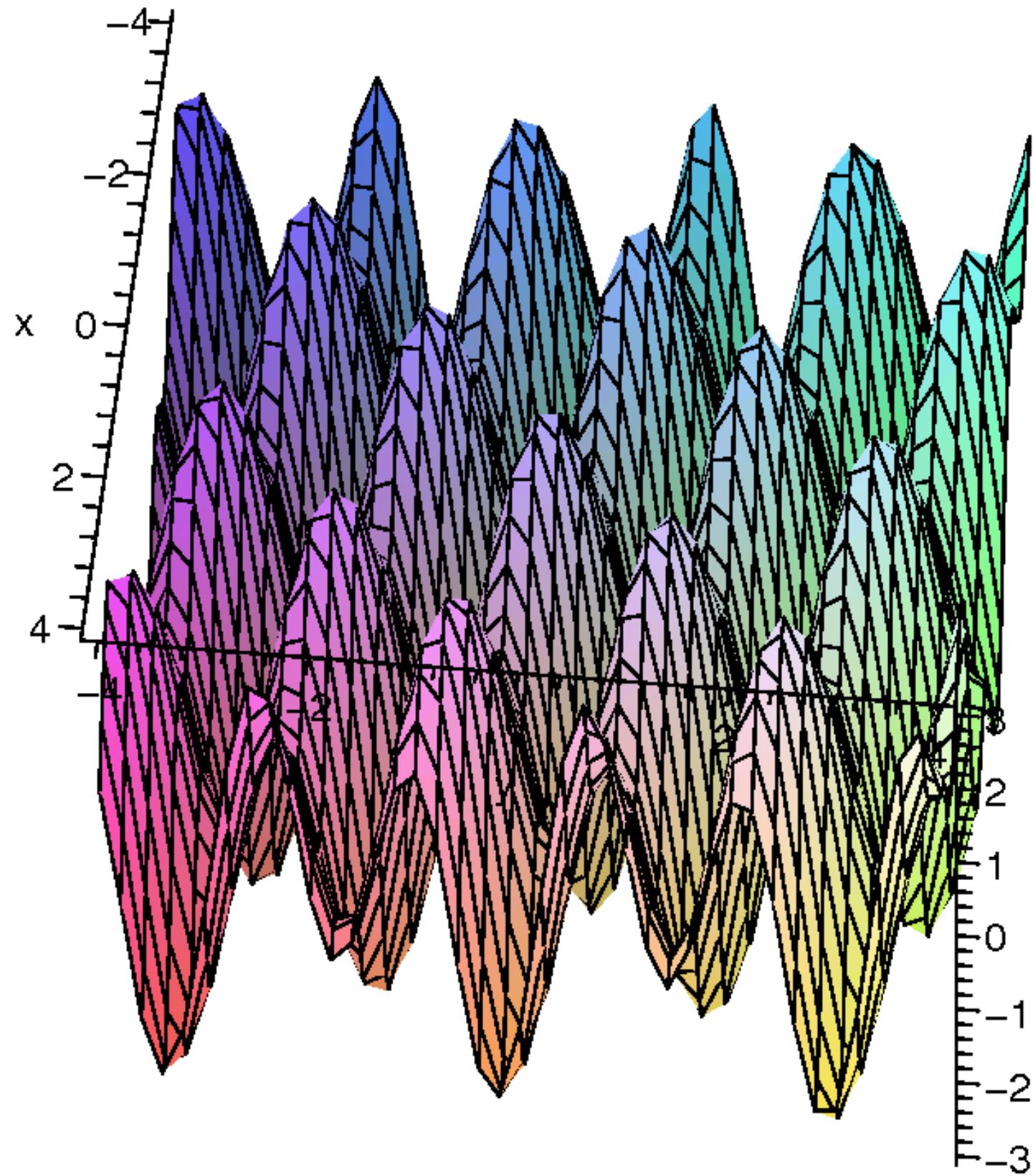


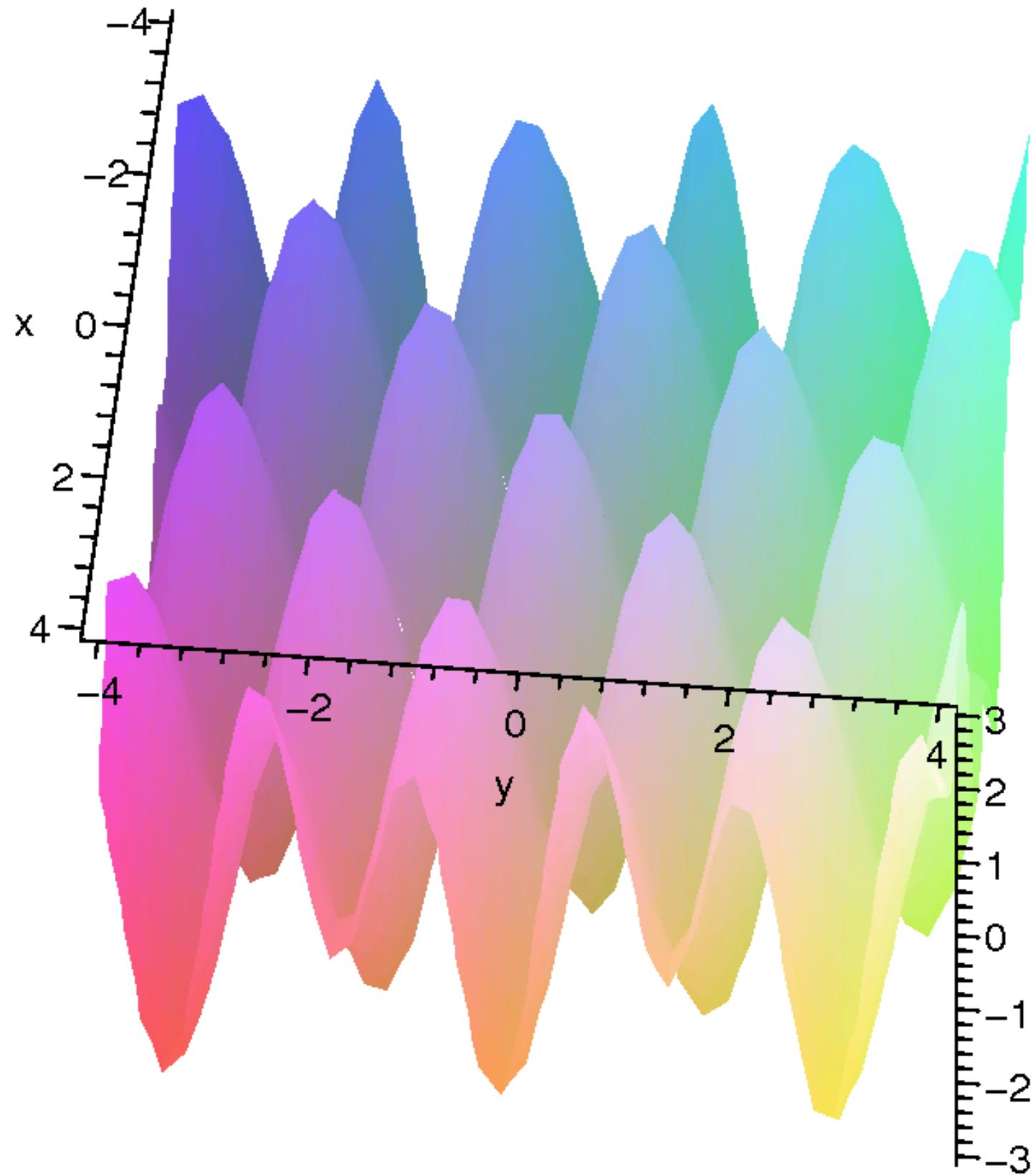




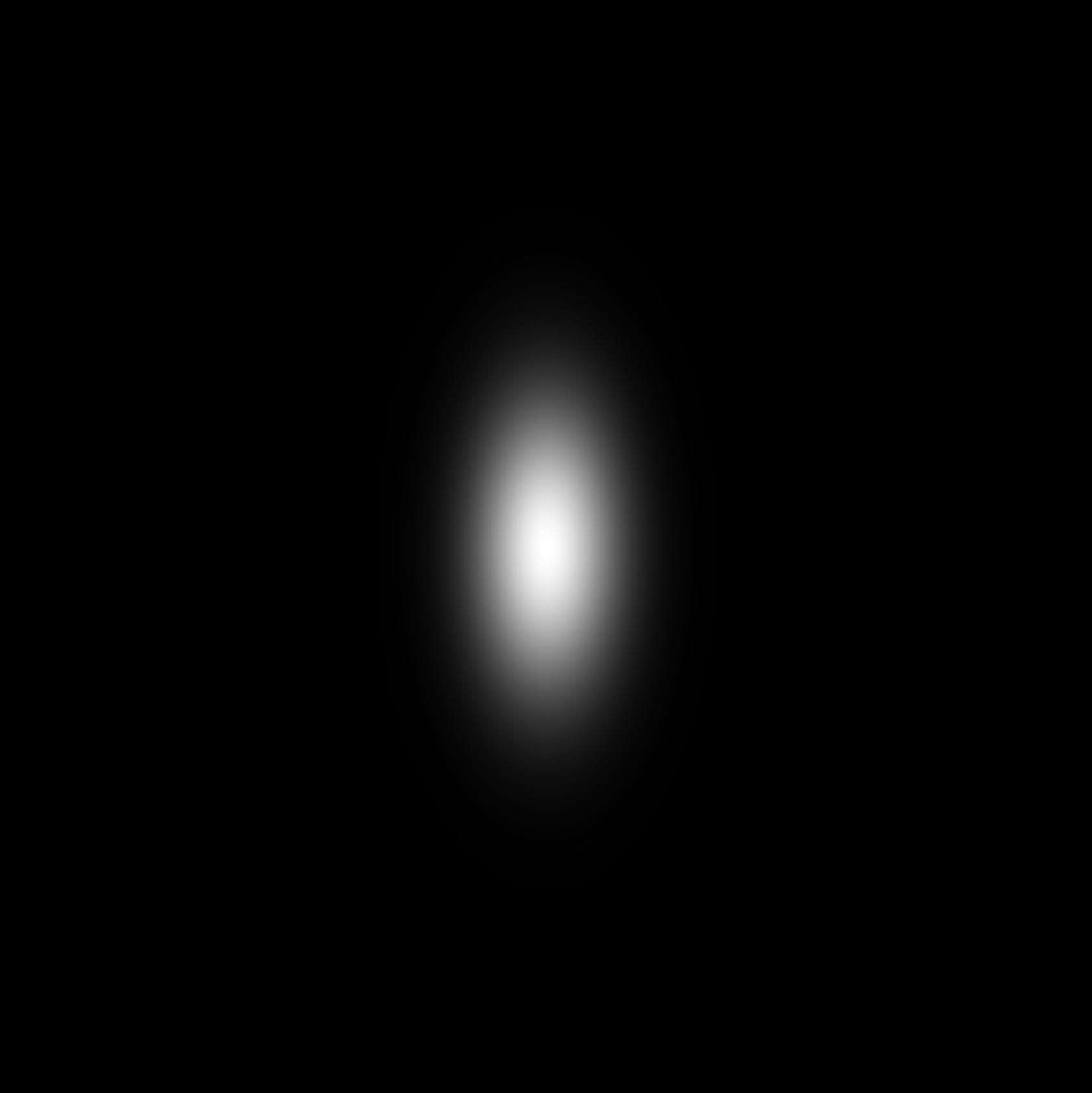


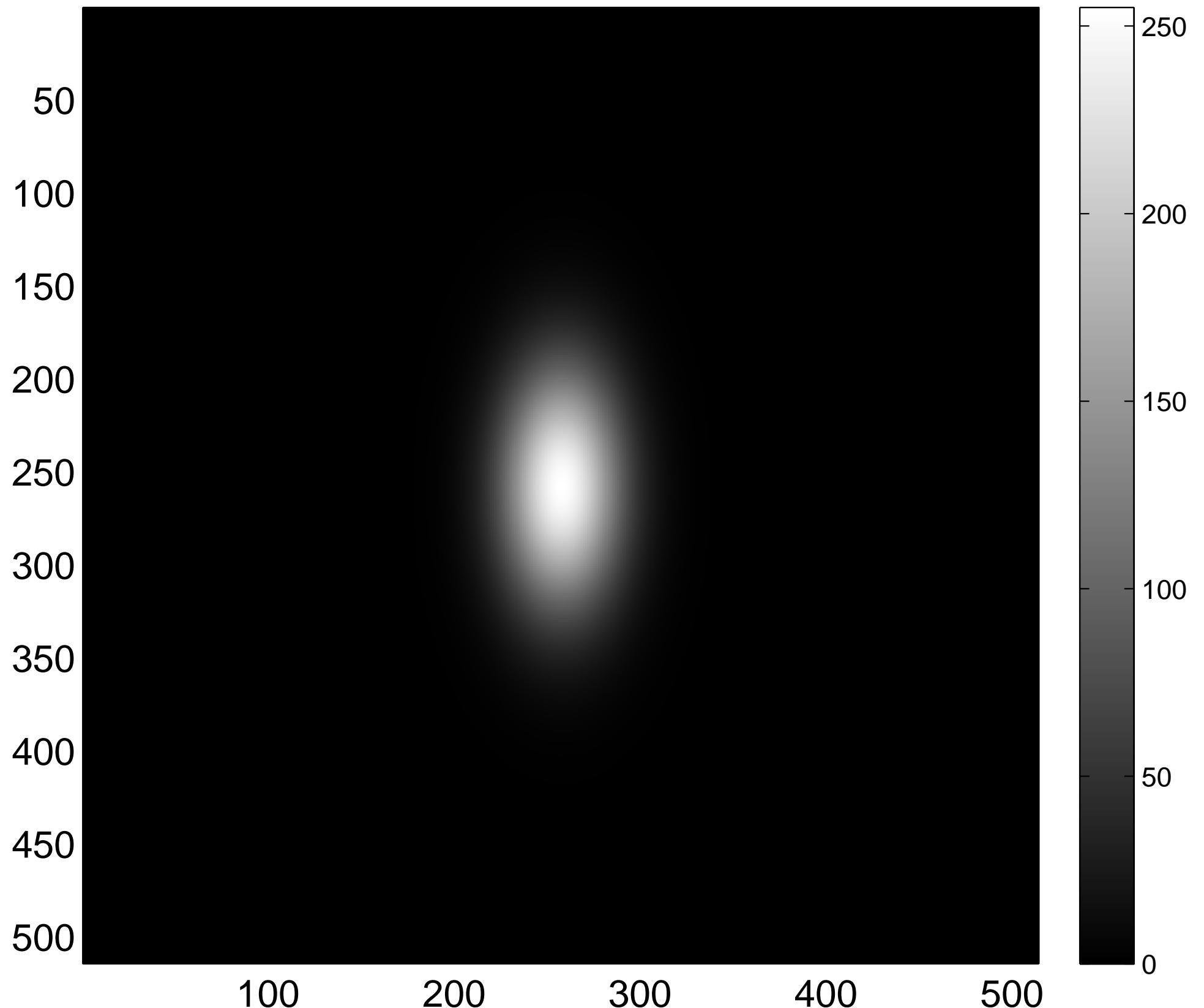




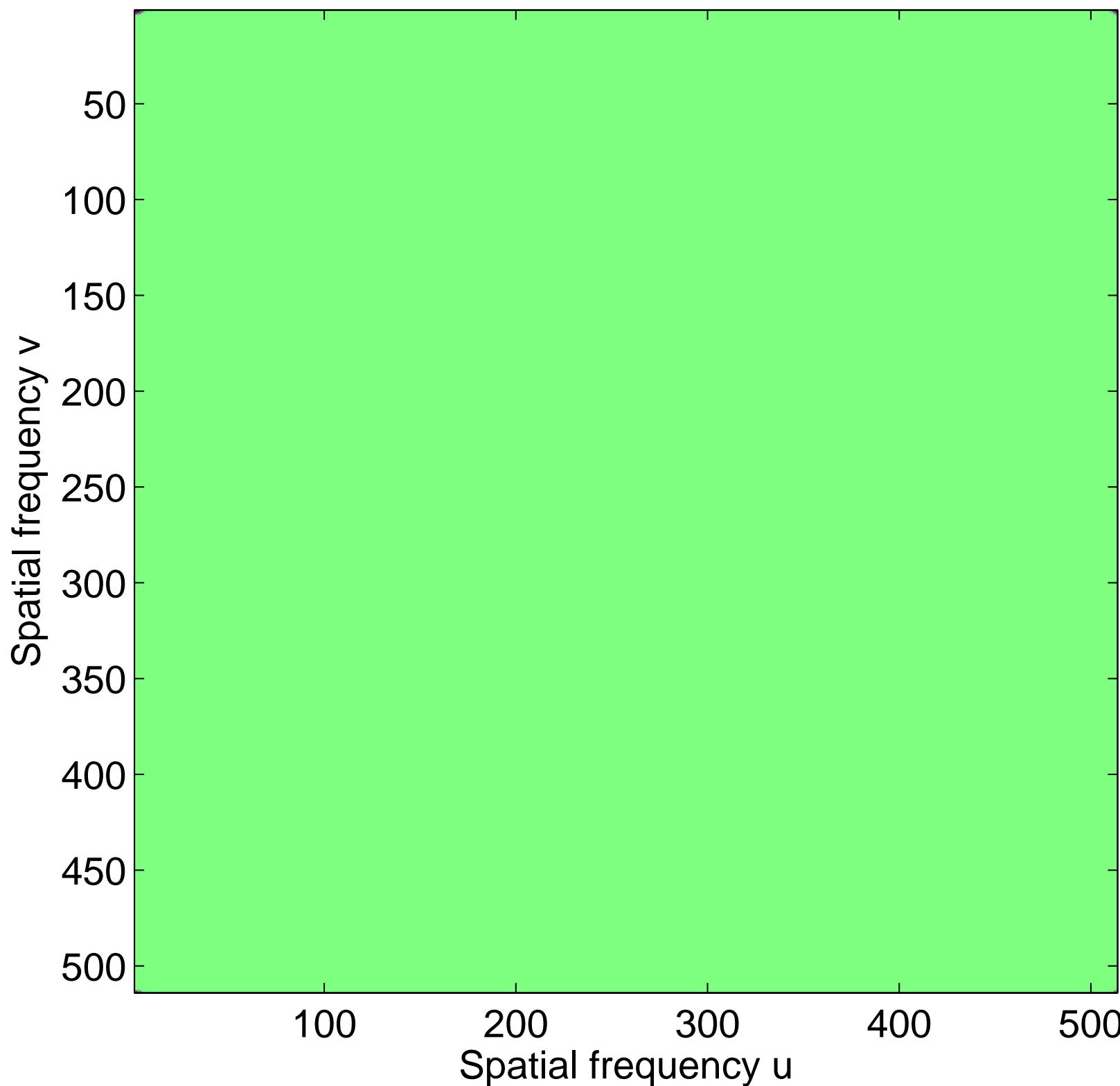




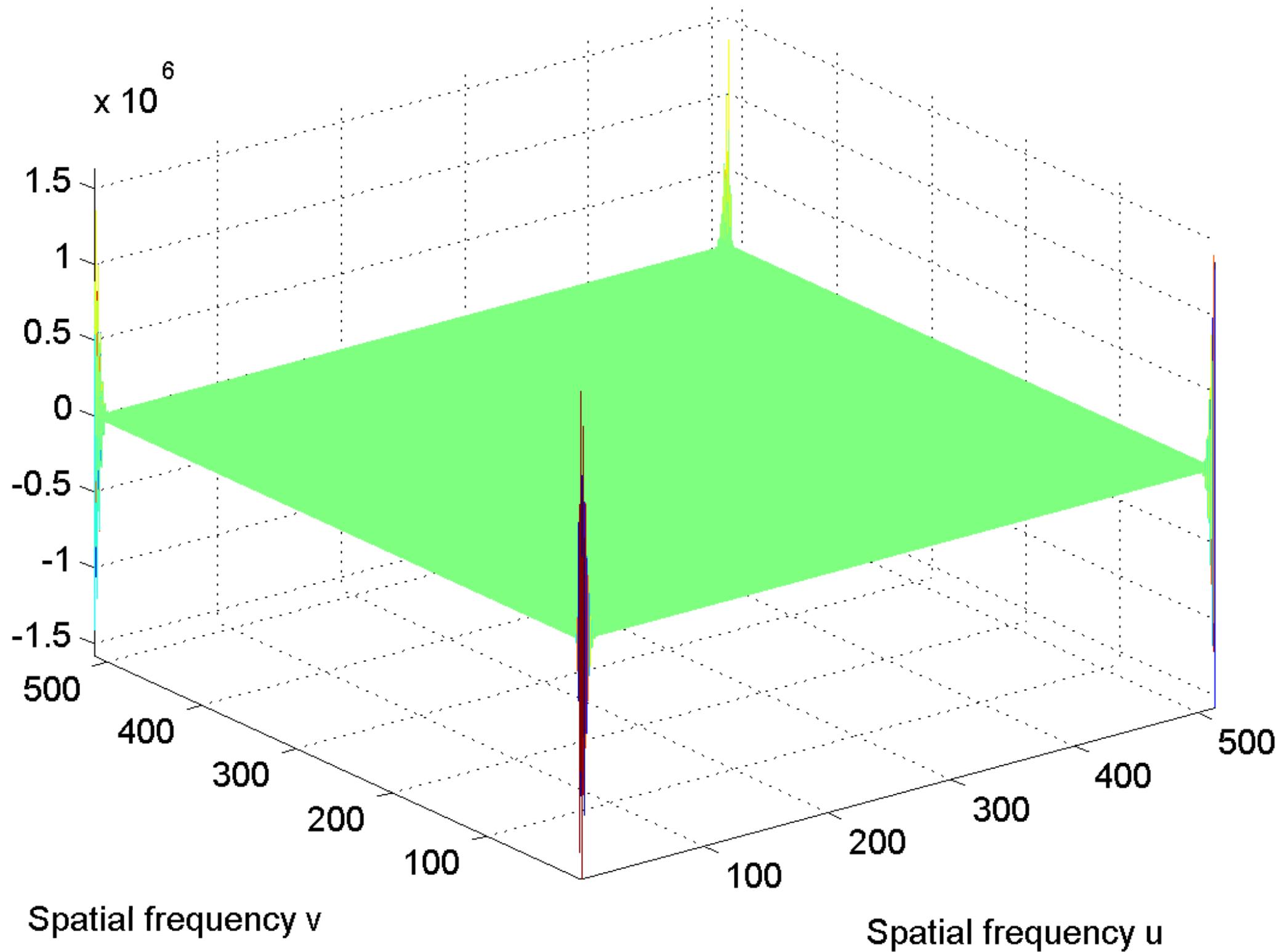




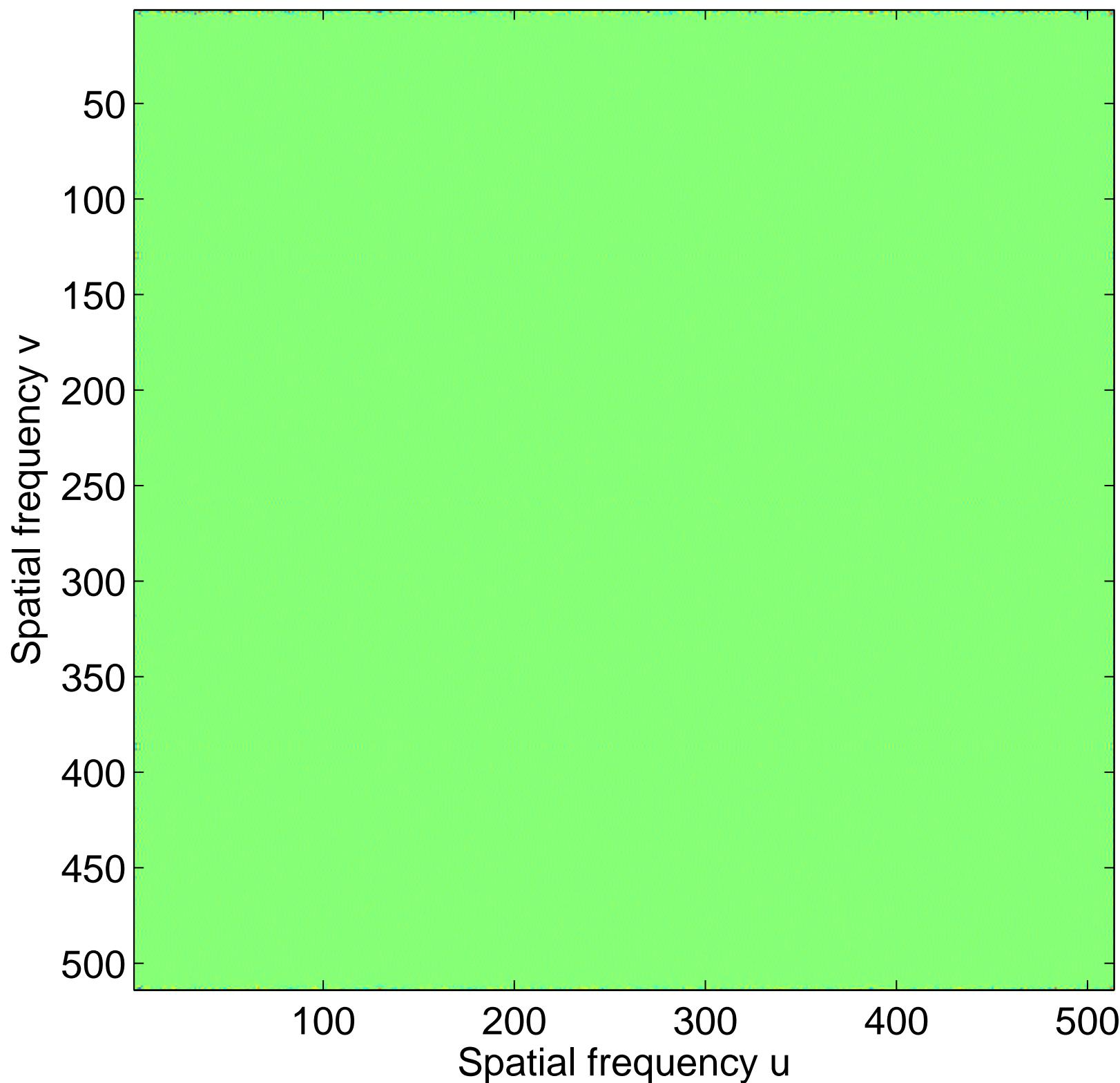
Real part of the spectrum



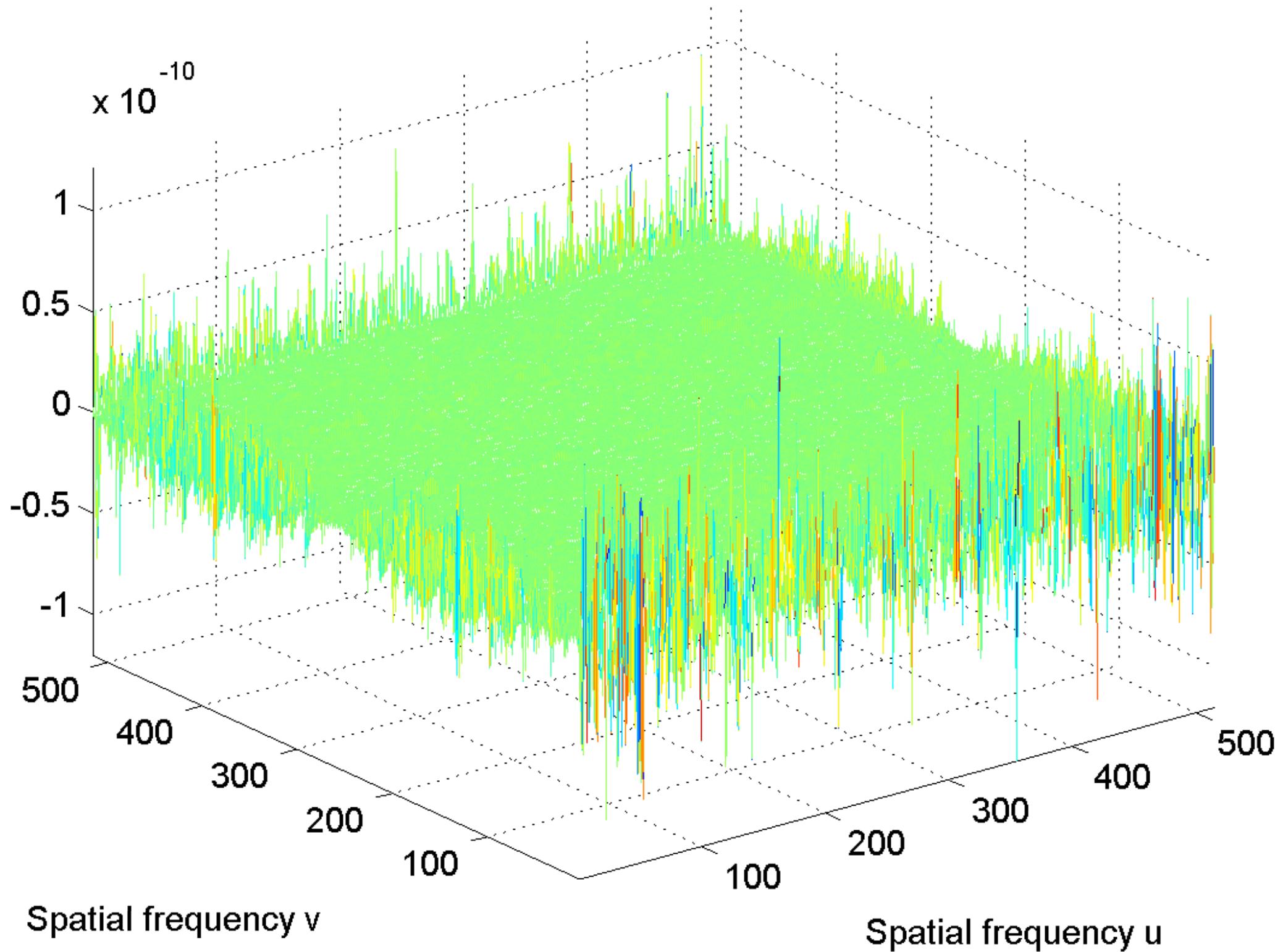
Real part of the spectrum



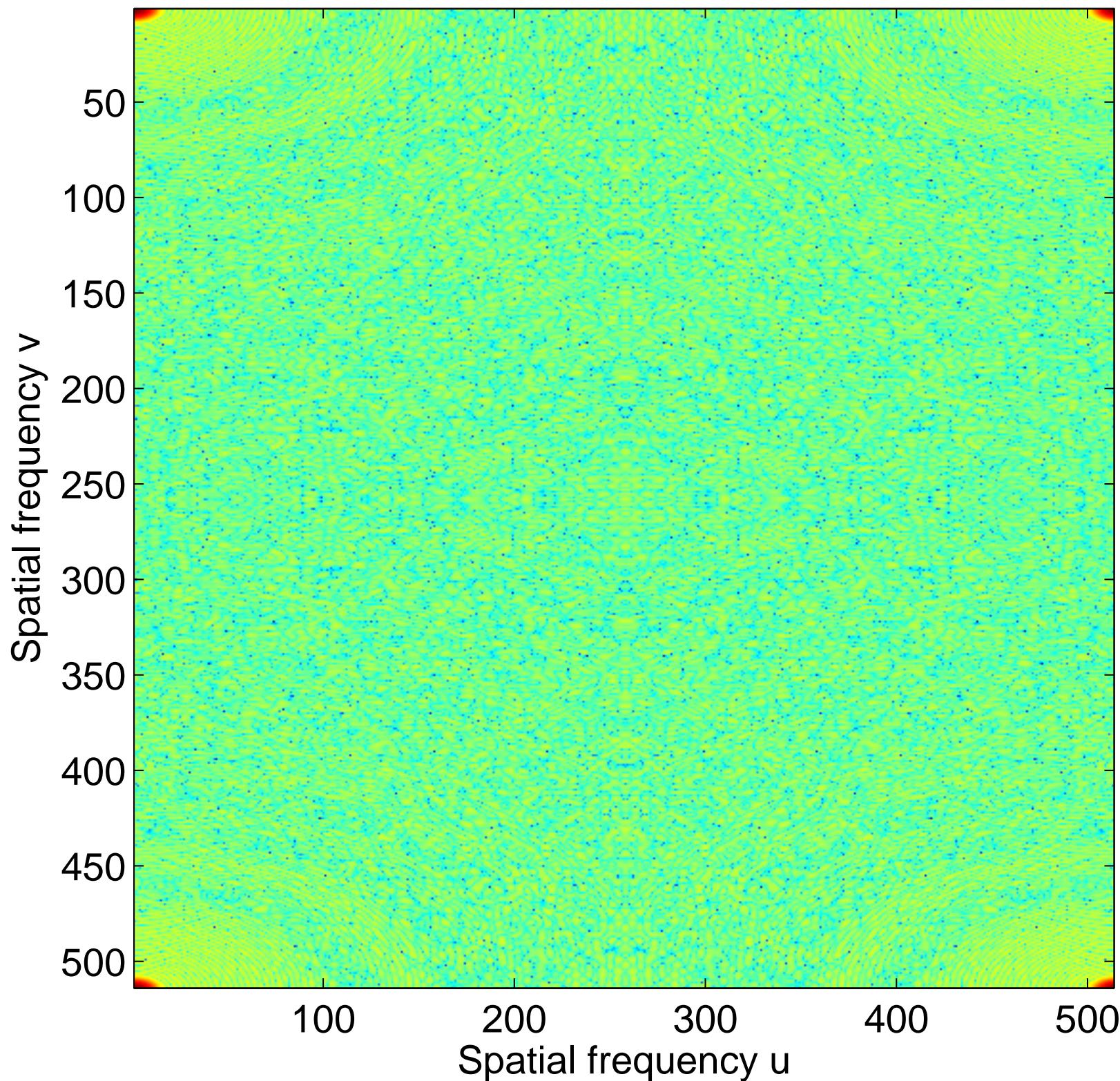
Imaginary part of the spectrum



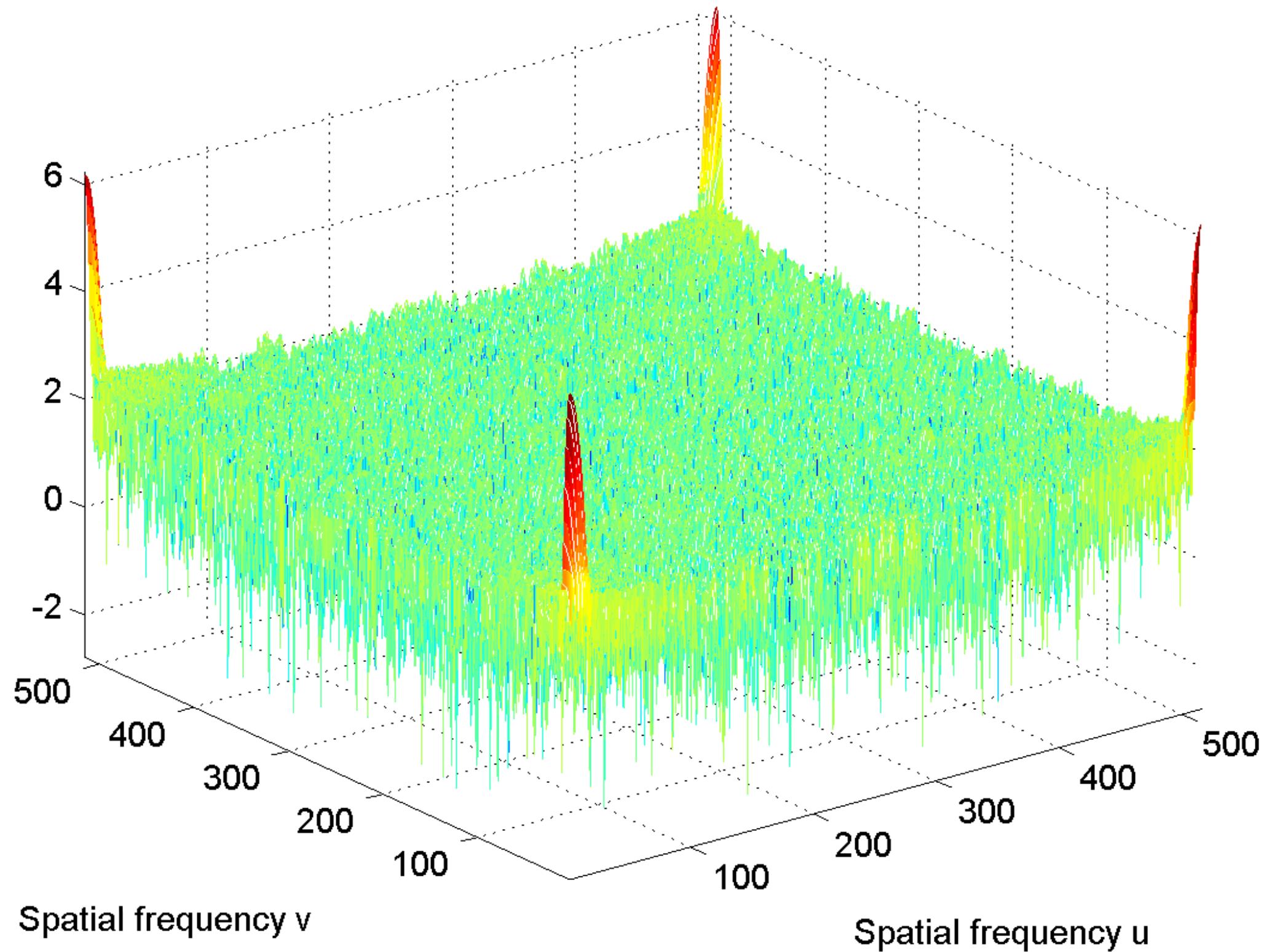
Imaginary part of the spectrum



log power spectrum

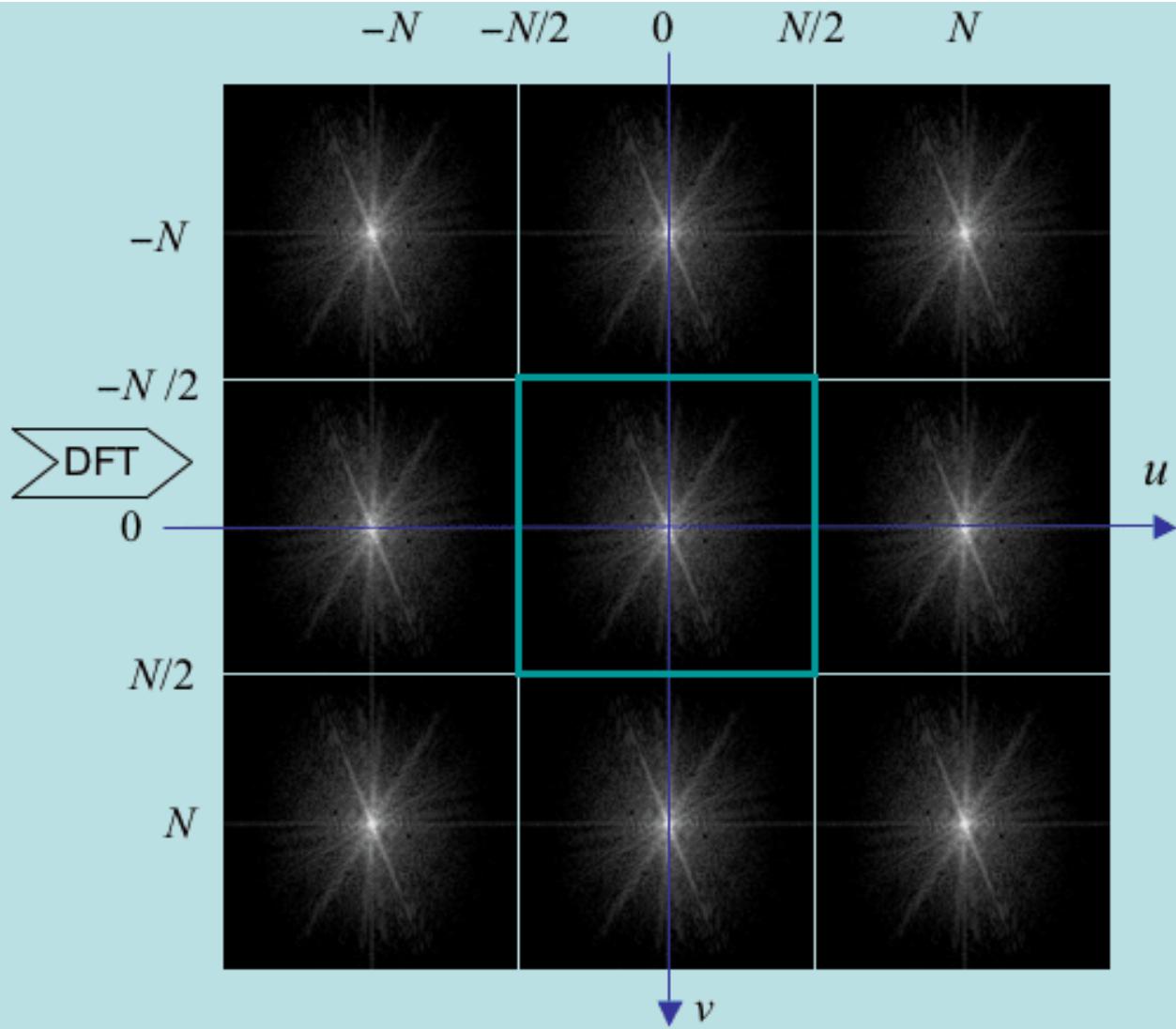


log power spectrum



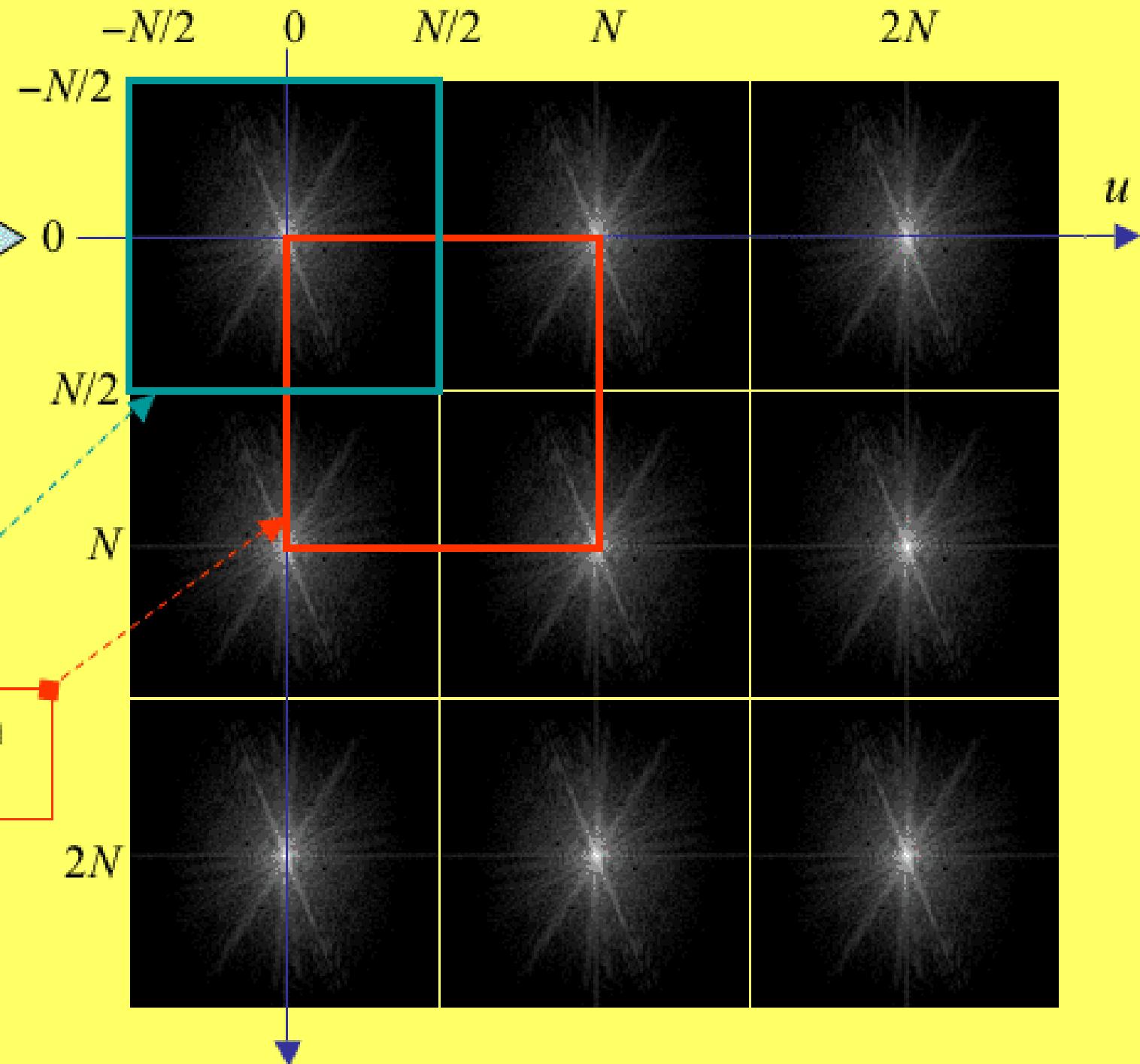


Spatial discontinuities caused by considering an image to be periodic



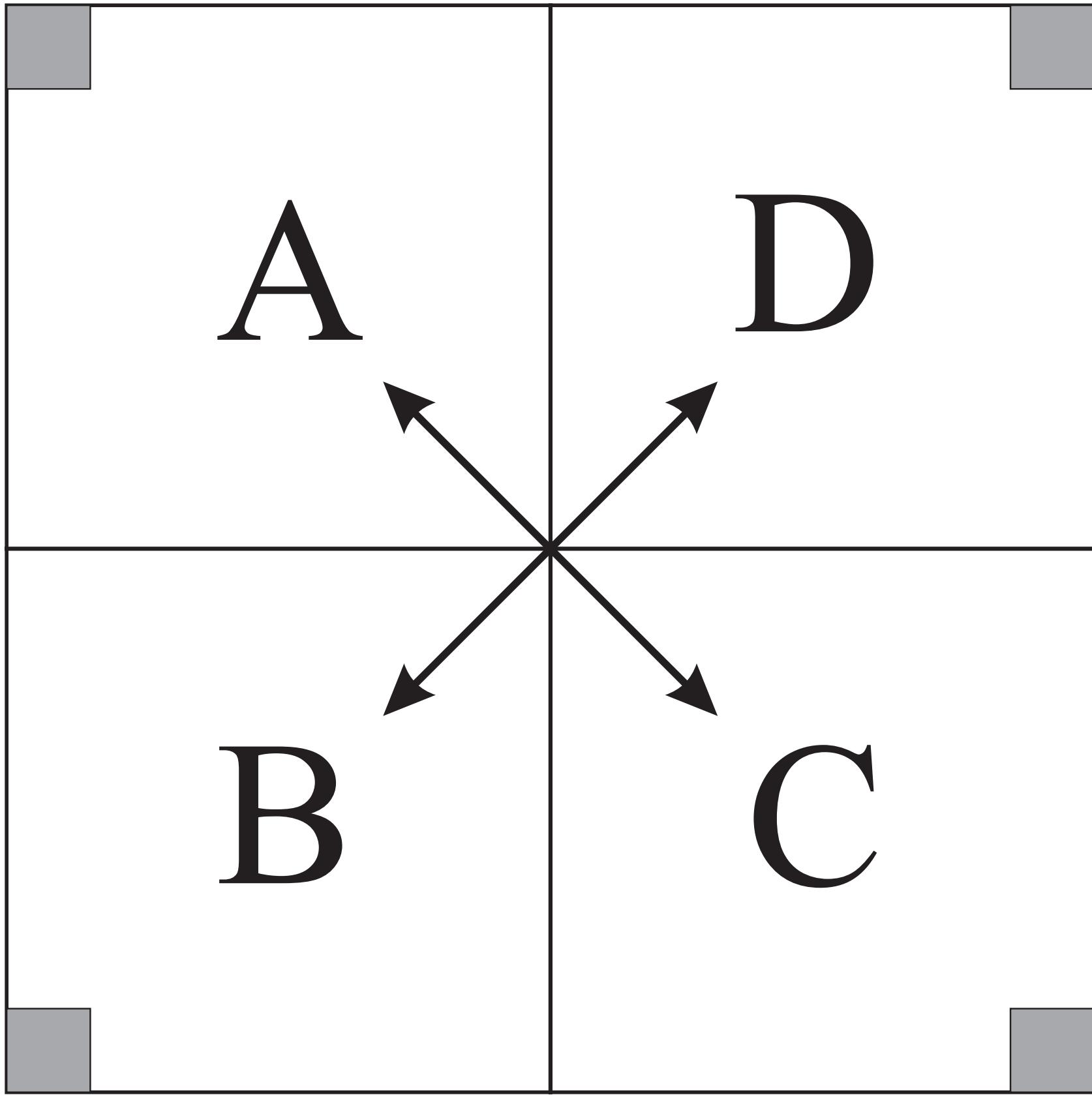


DFT



Representation of spectra
being easier to interpret

Single period of the spectrum
computed by a DFT

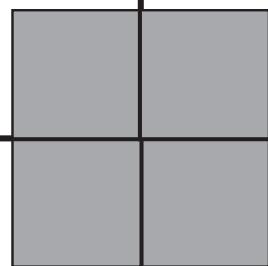


C

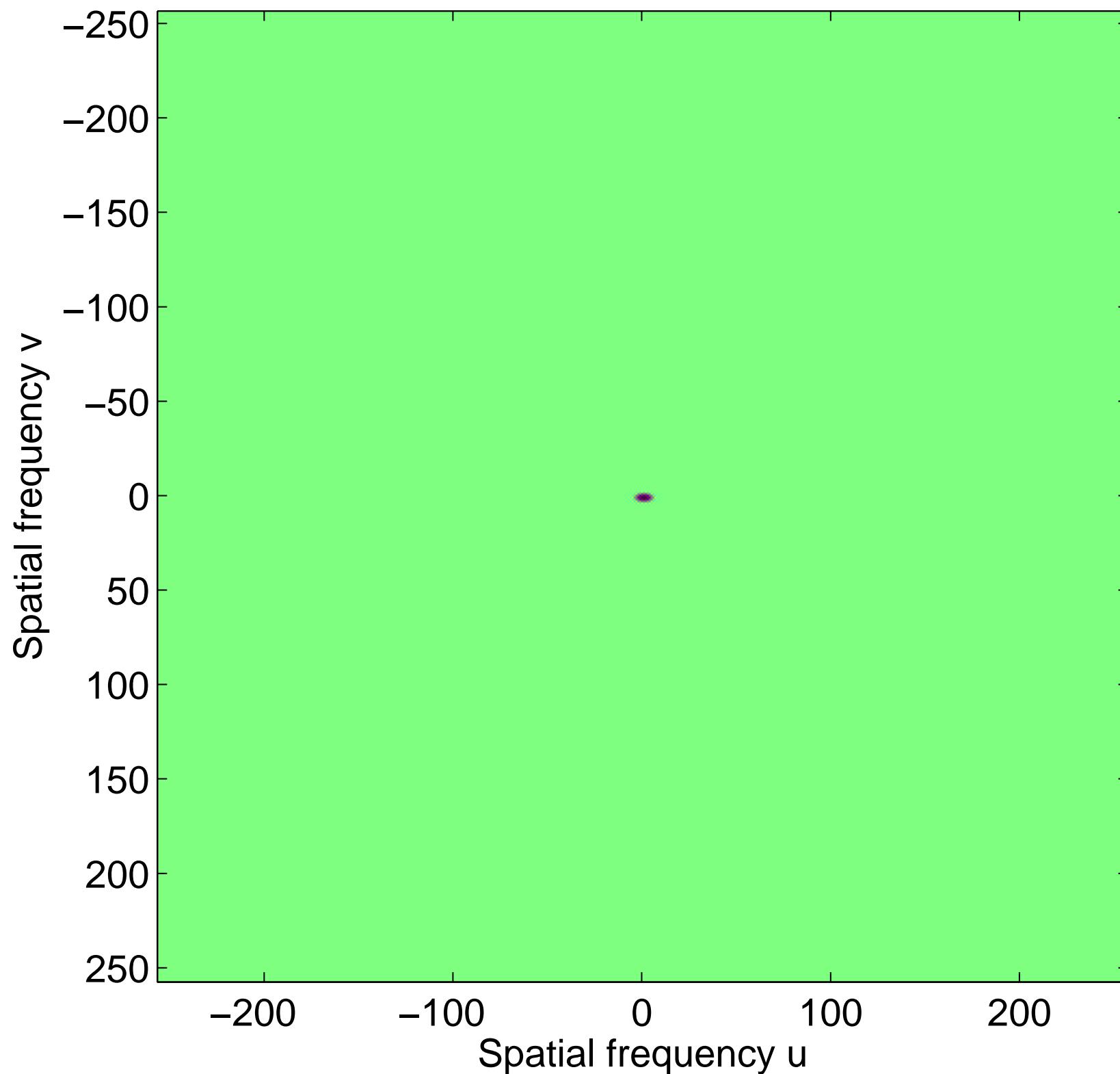
B

D

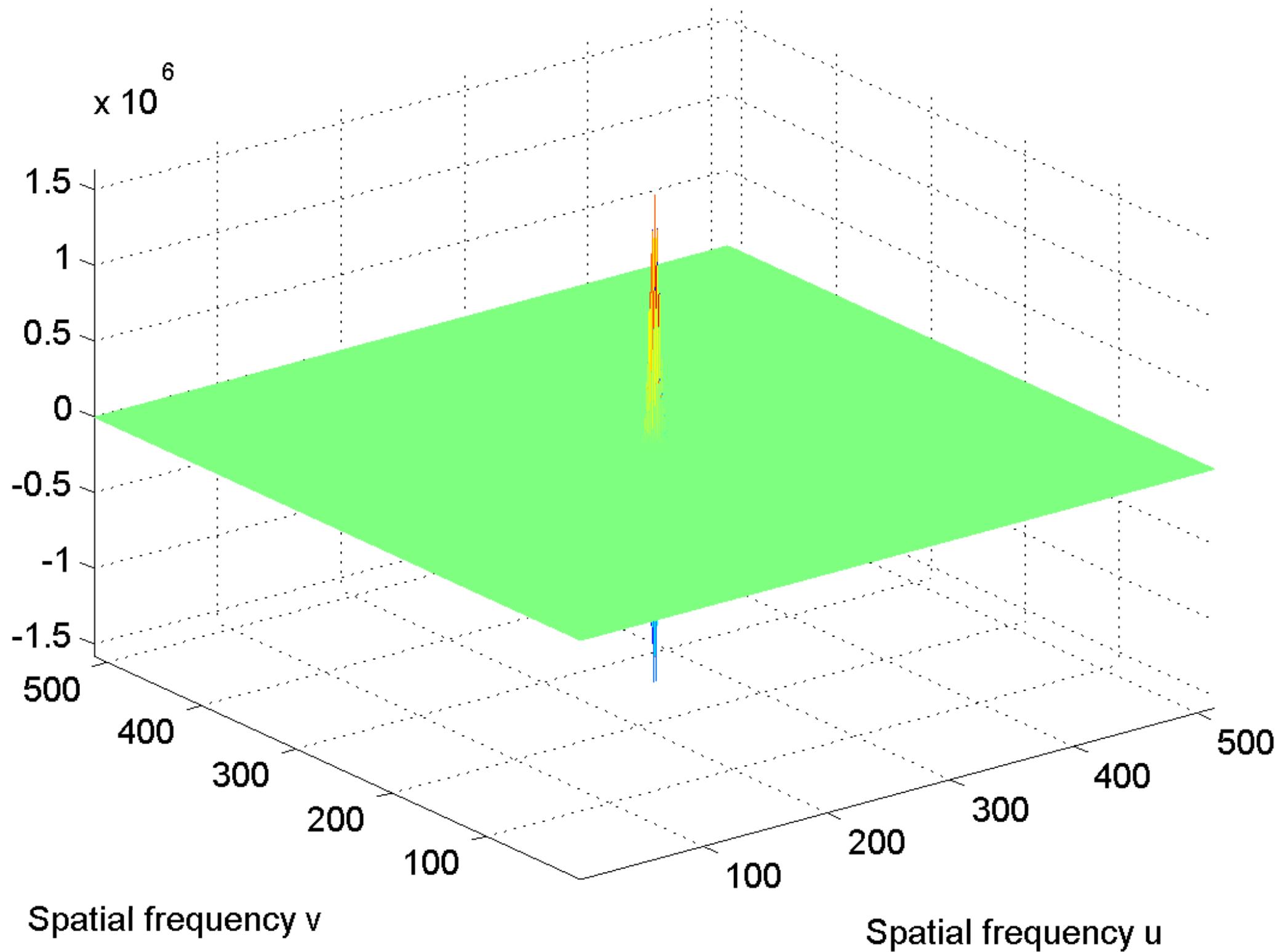
A



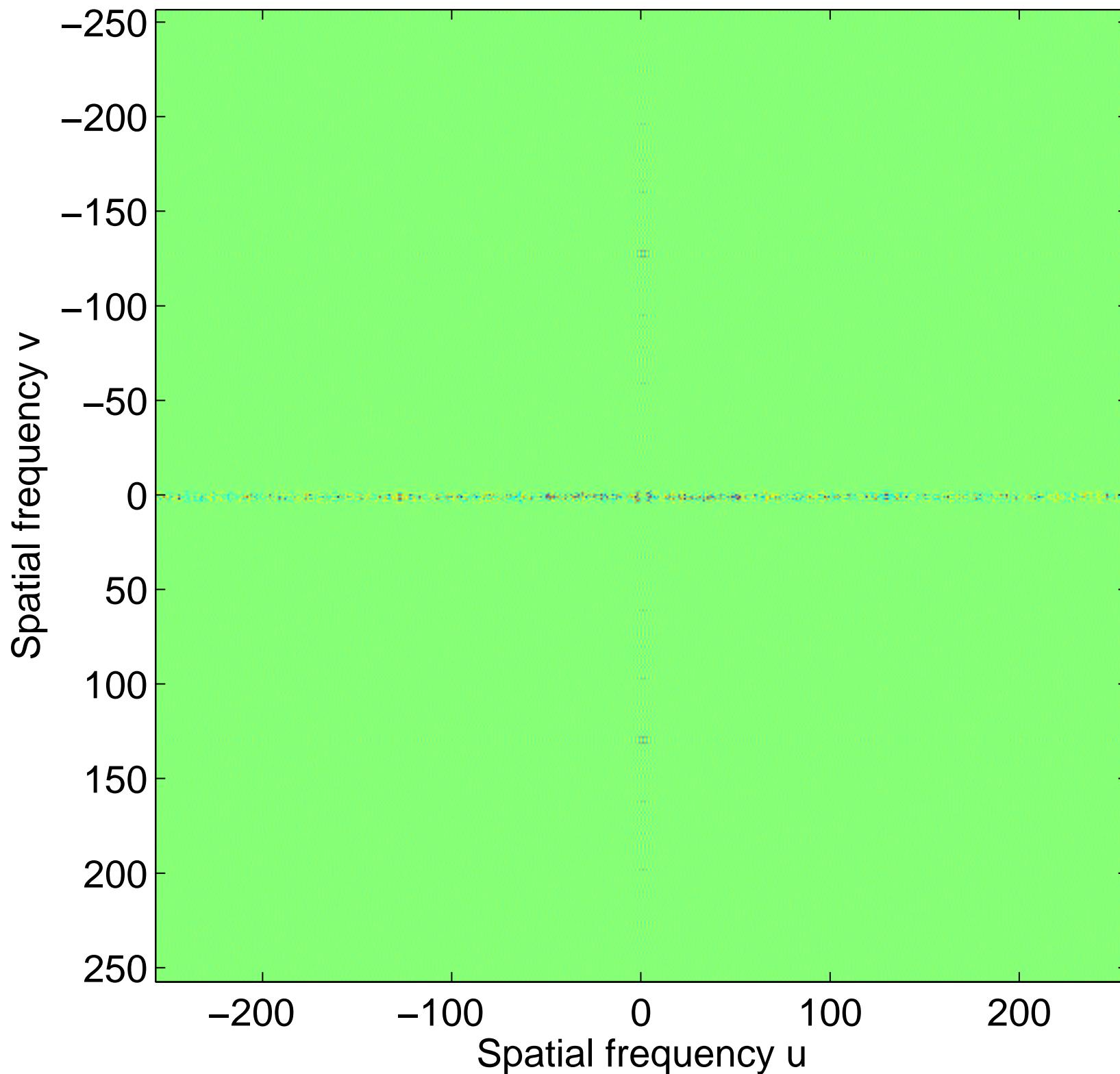
Real part of the spectrum, centered



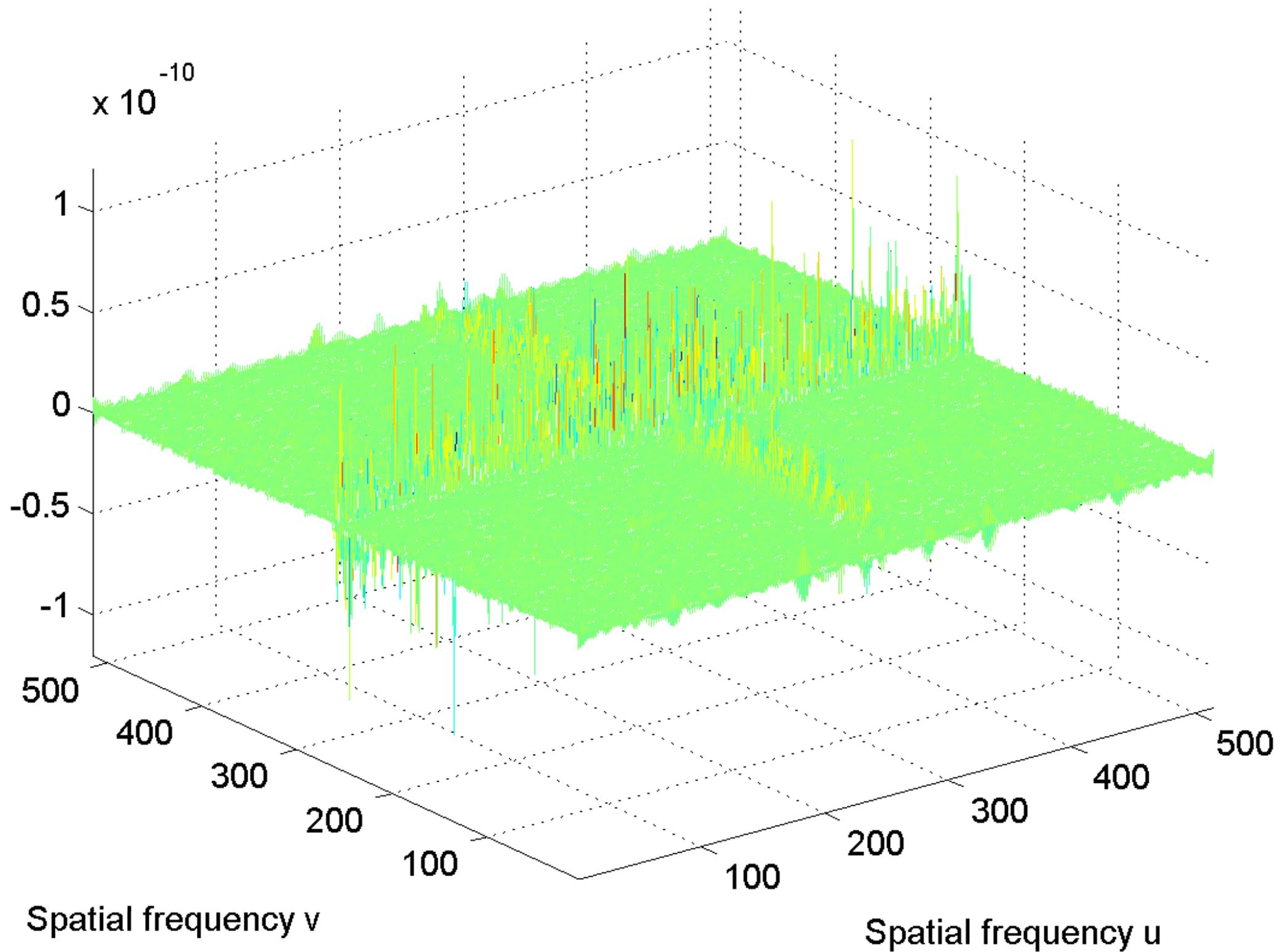
Real part of the spectrum, centered



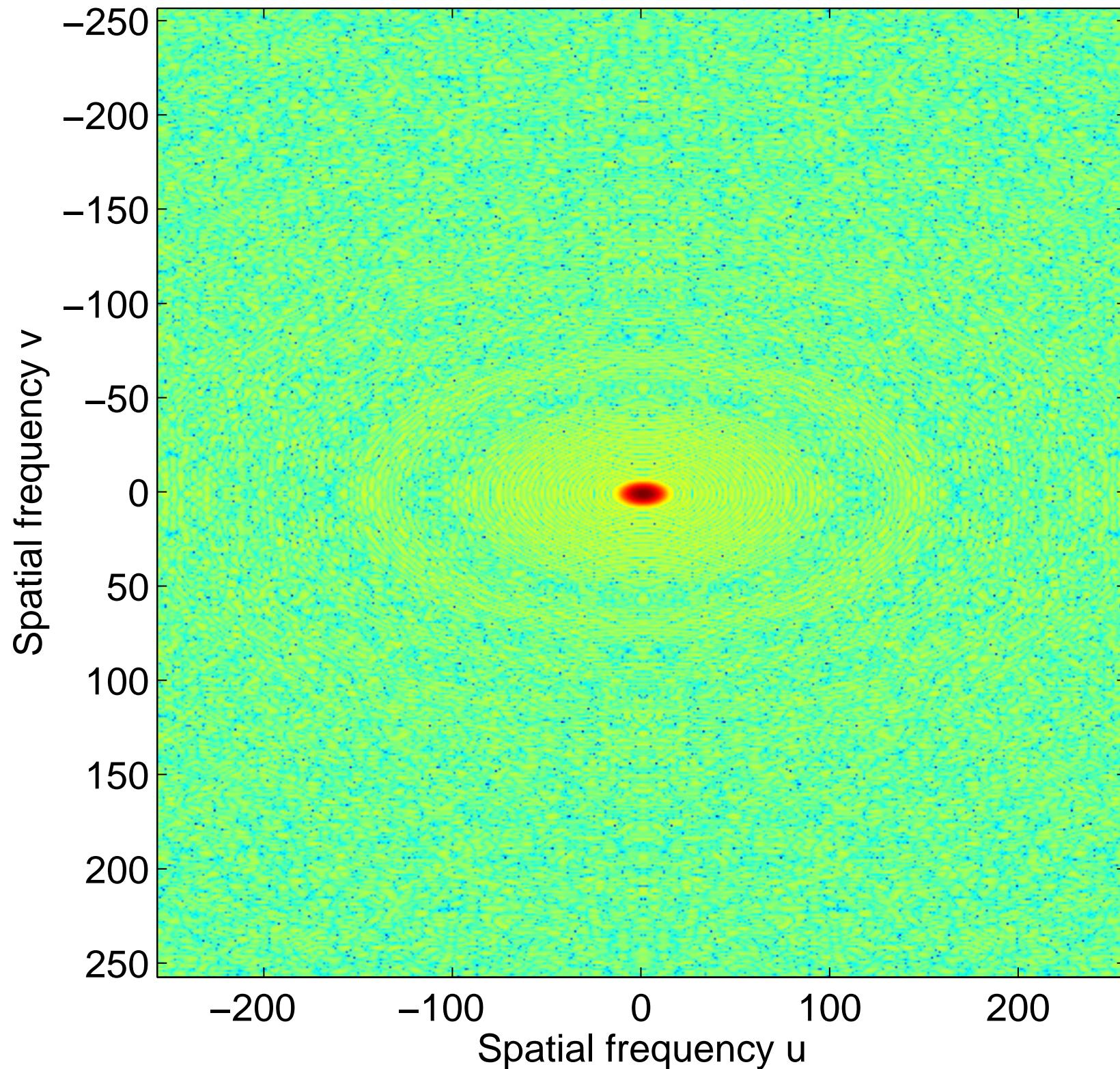
Imaginary part of the spectrum, centered



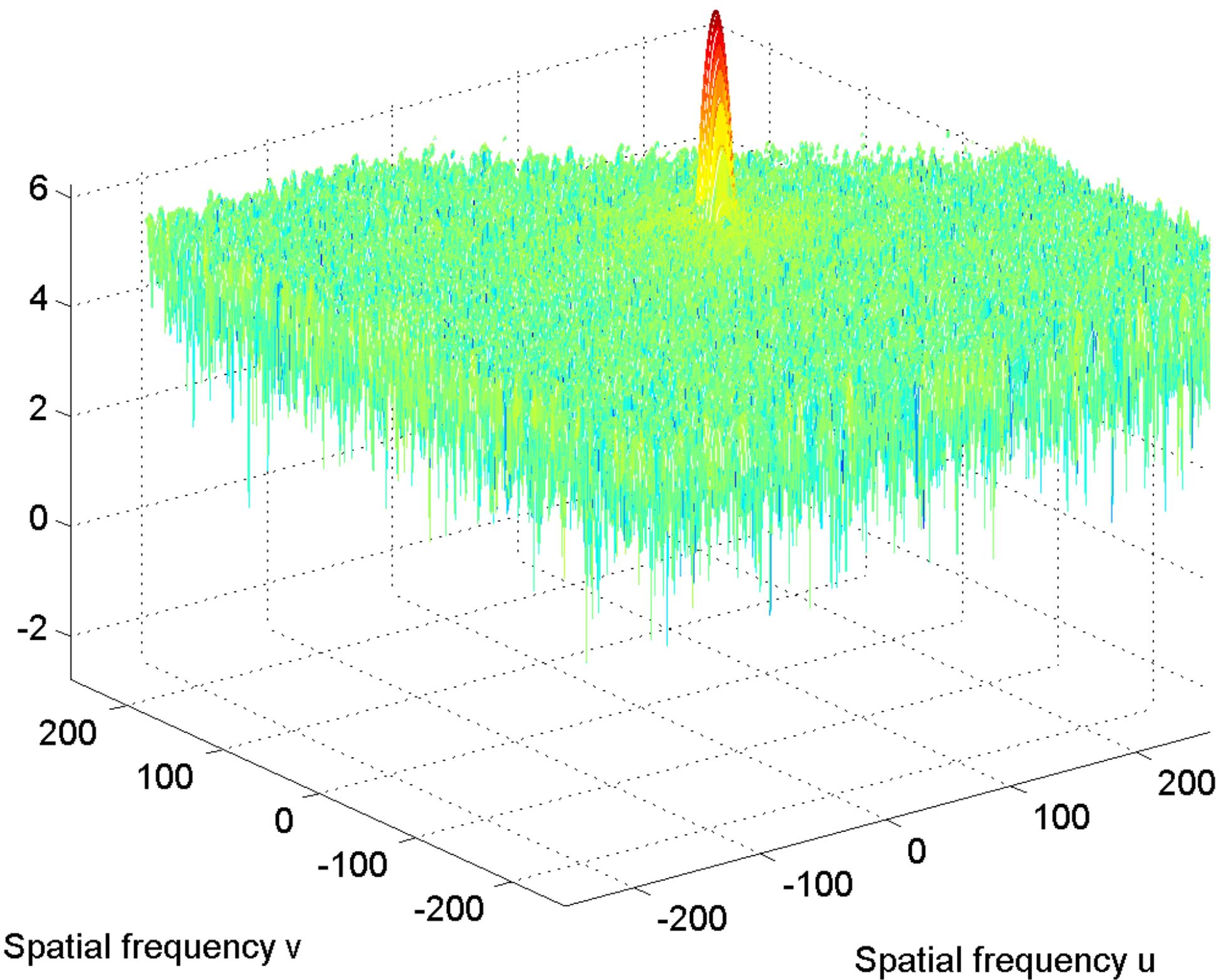
Imaginary part of the spectrum, centered

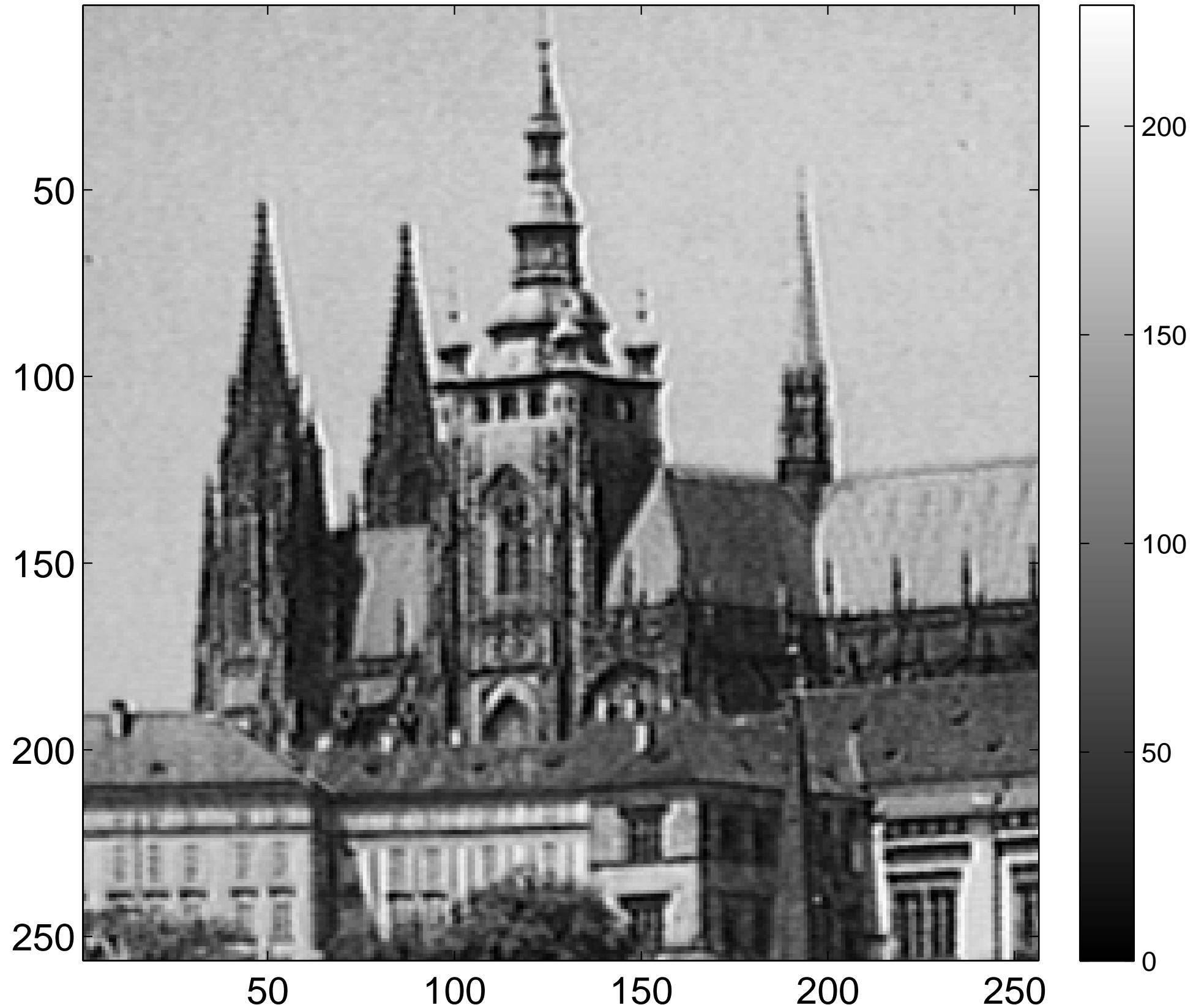


log power spectrum, centered

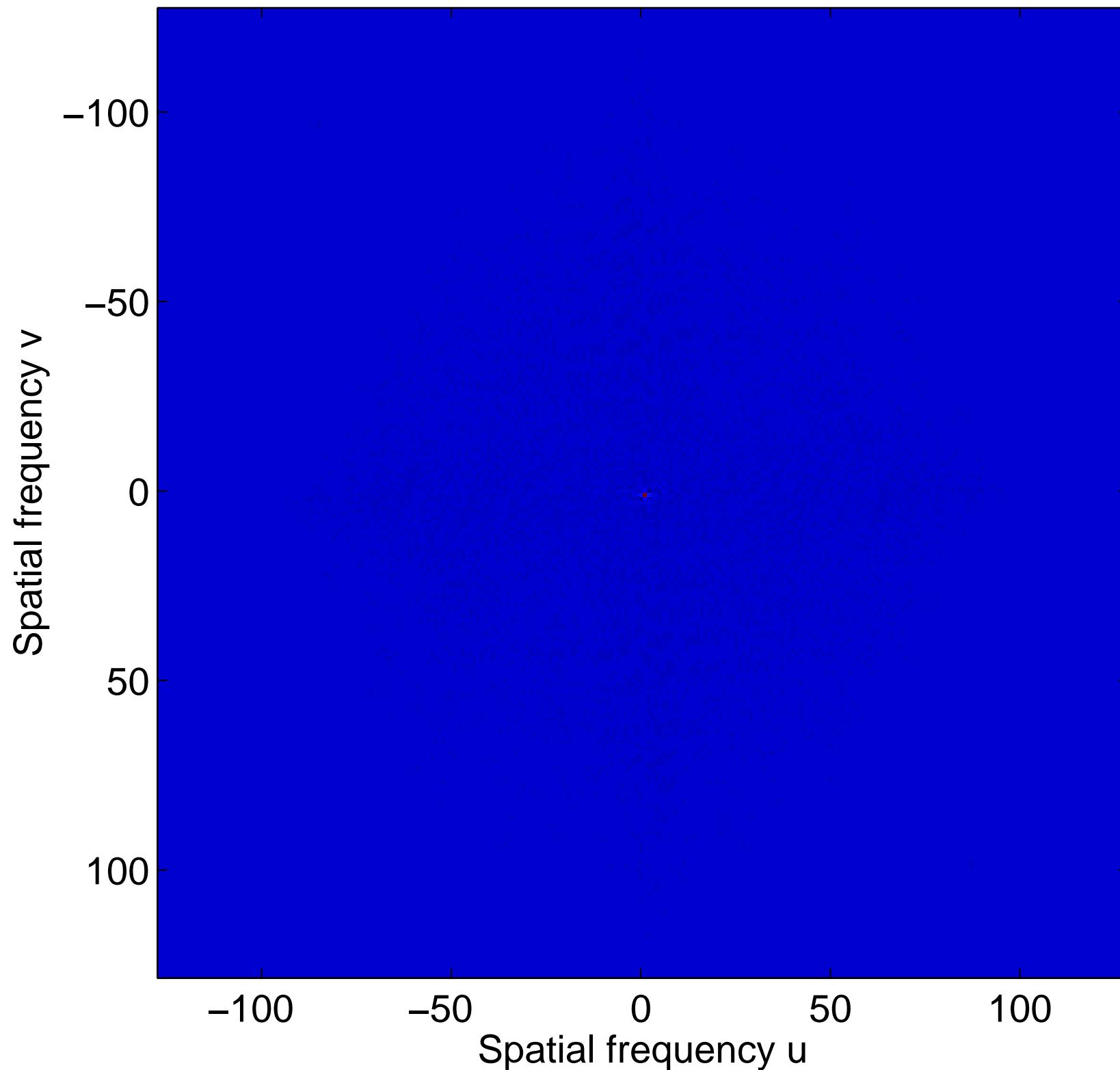


log power spectrum, centered

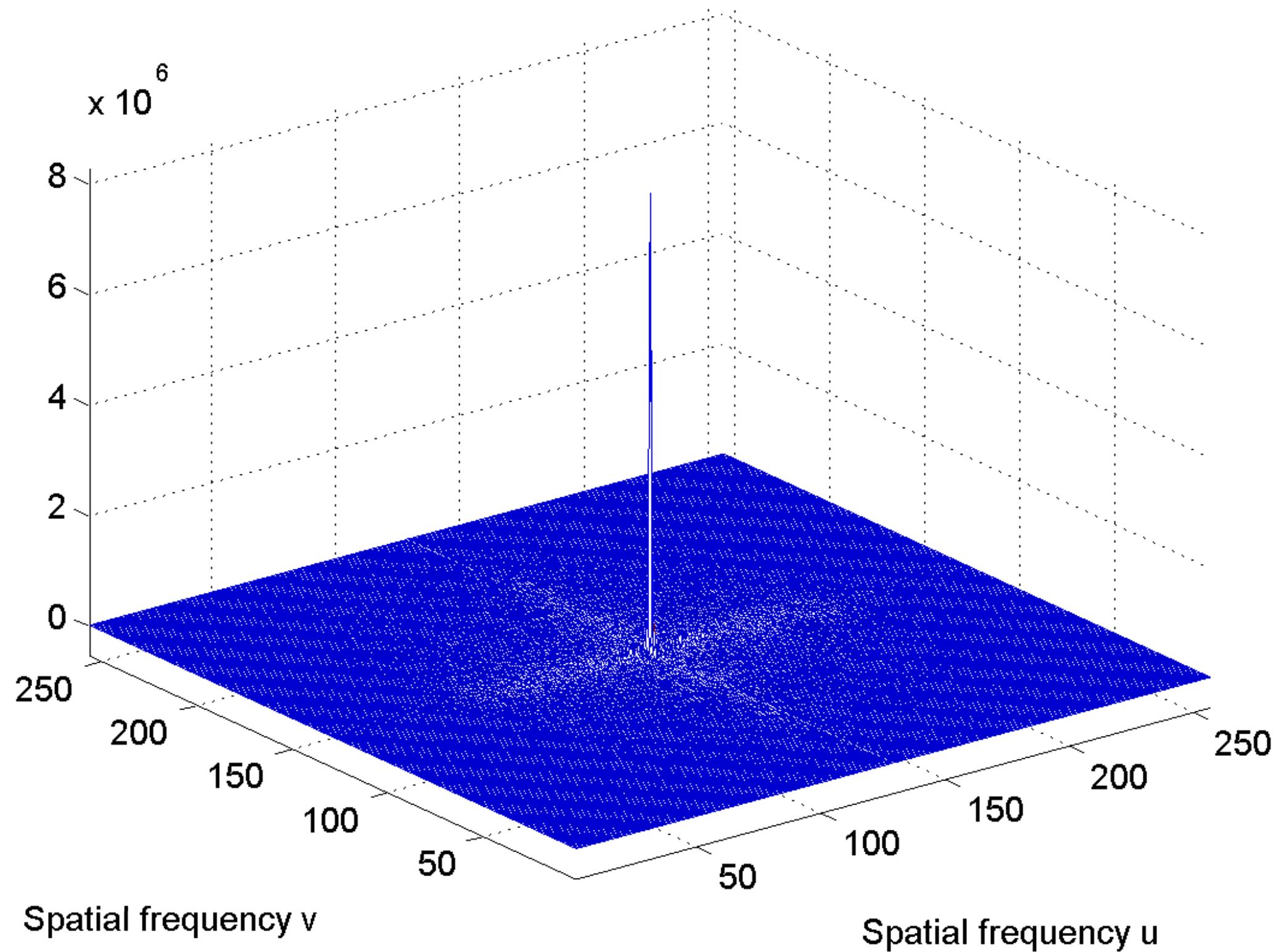




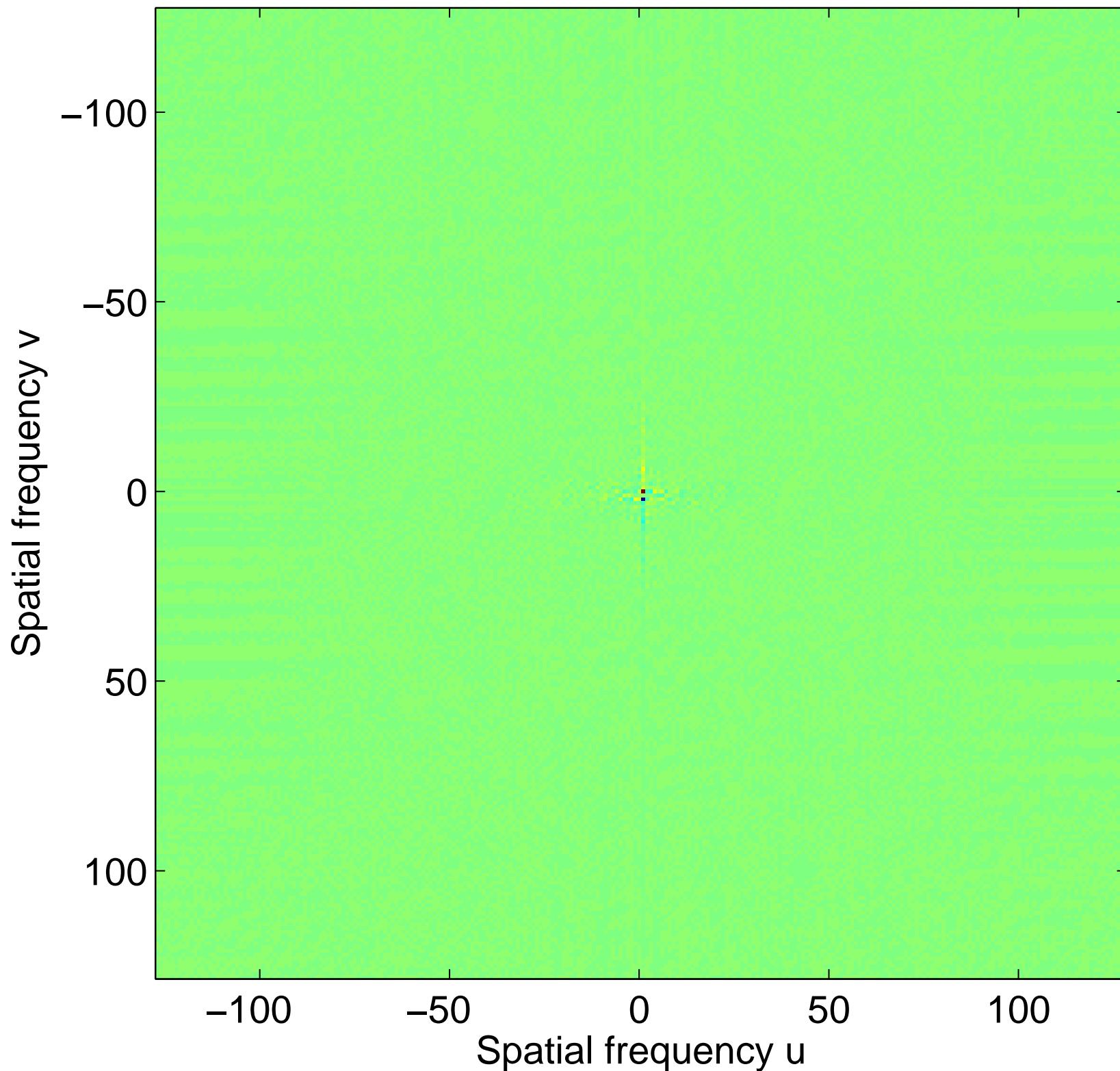
Real part of the spectrum, centered



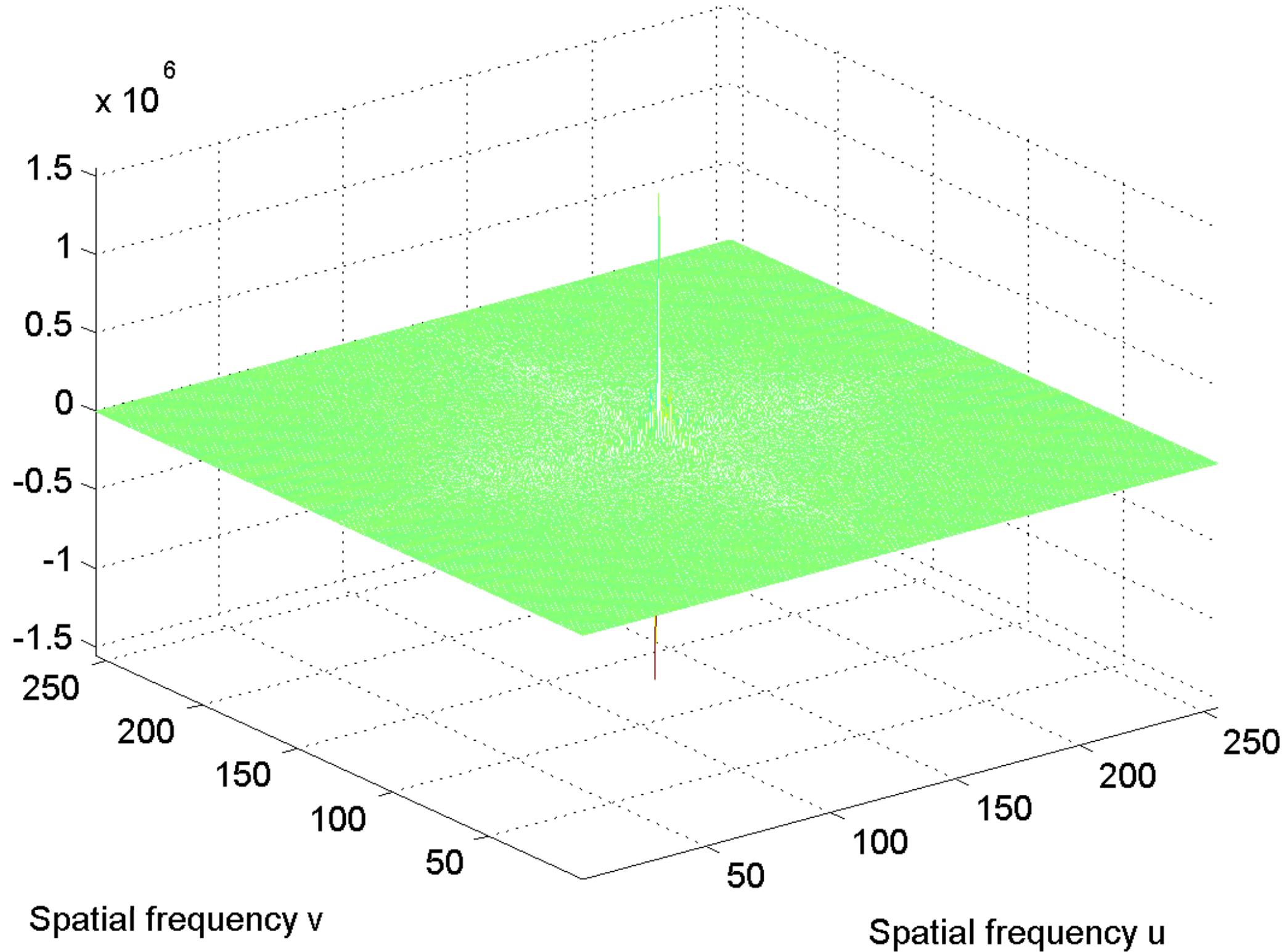
Real part of the spectrum, centered



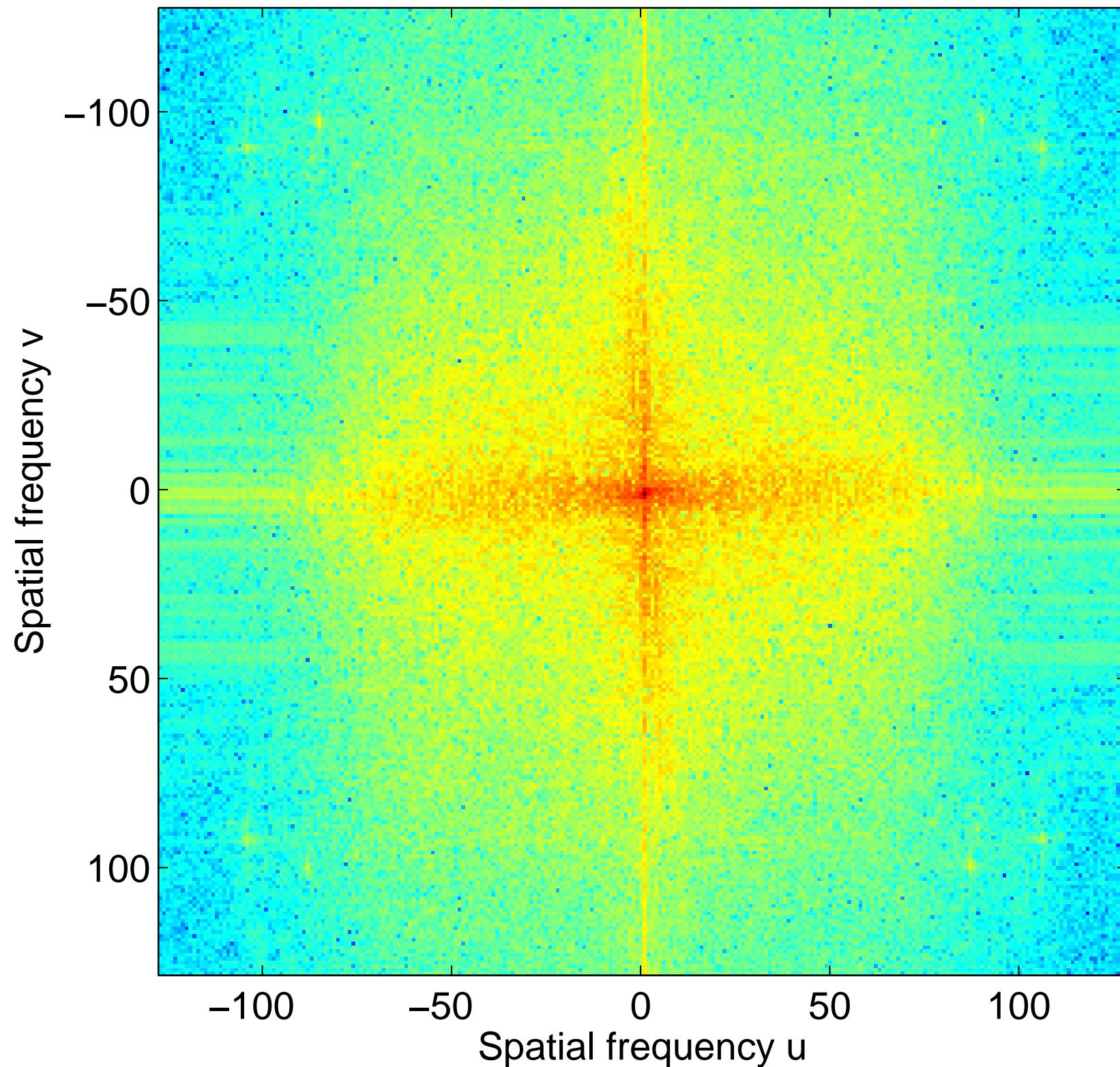
Imaginary part of the spectrum, centered



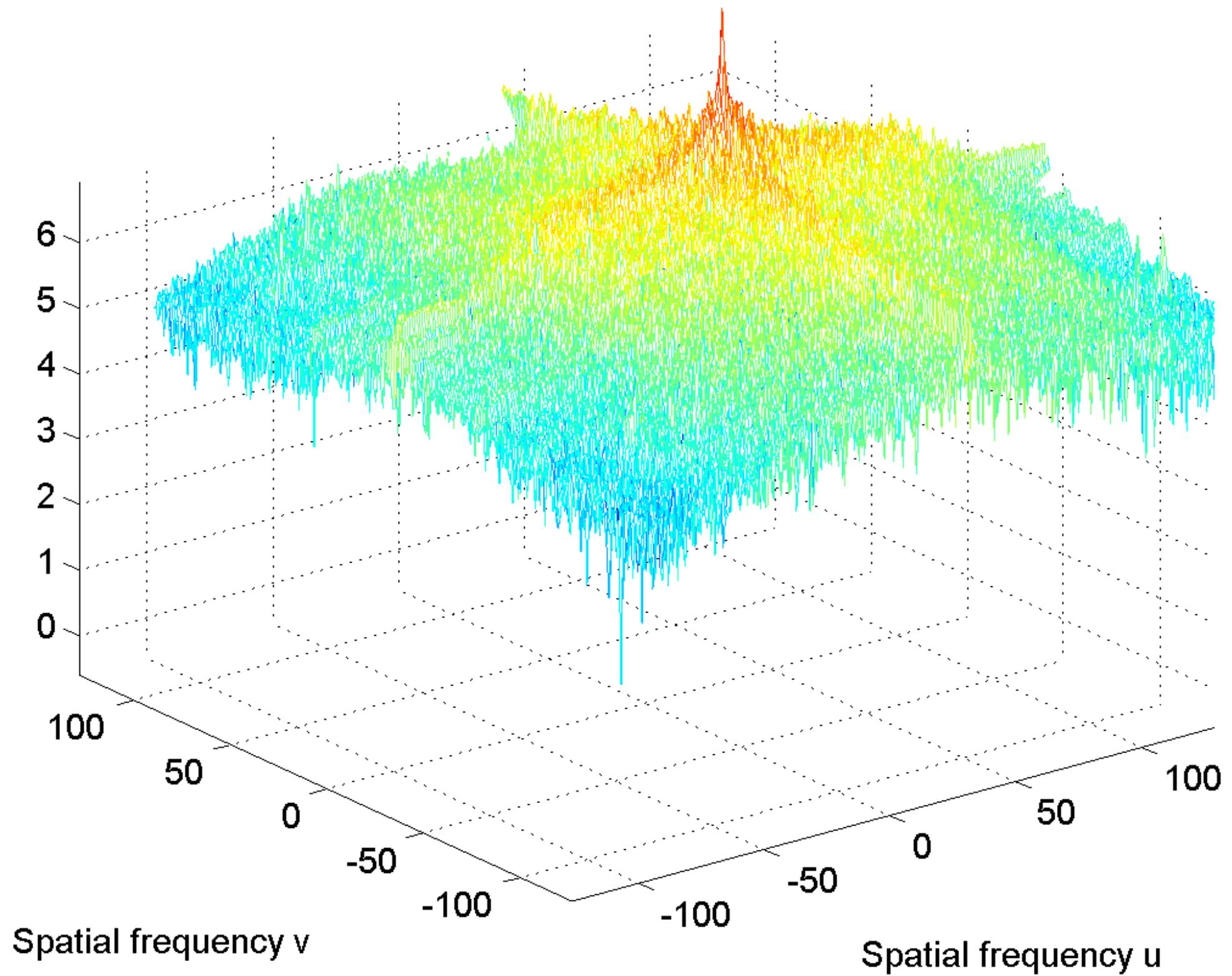
Imaginary part of the spectrum, centered

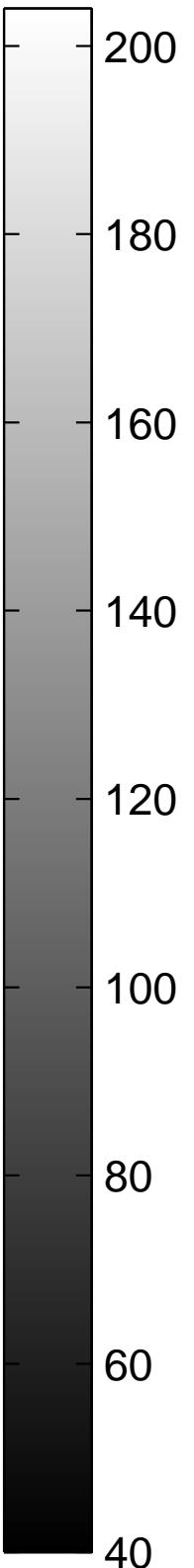
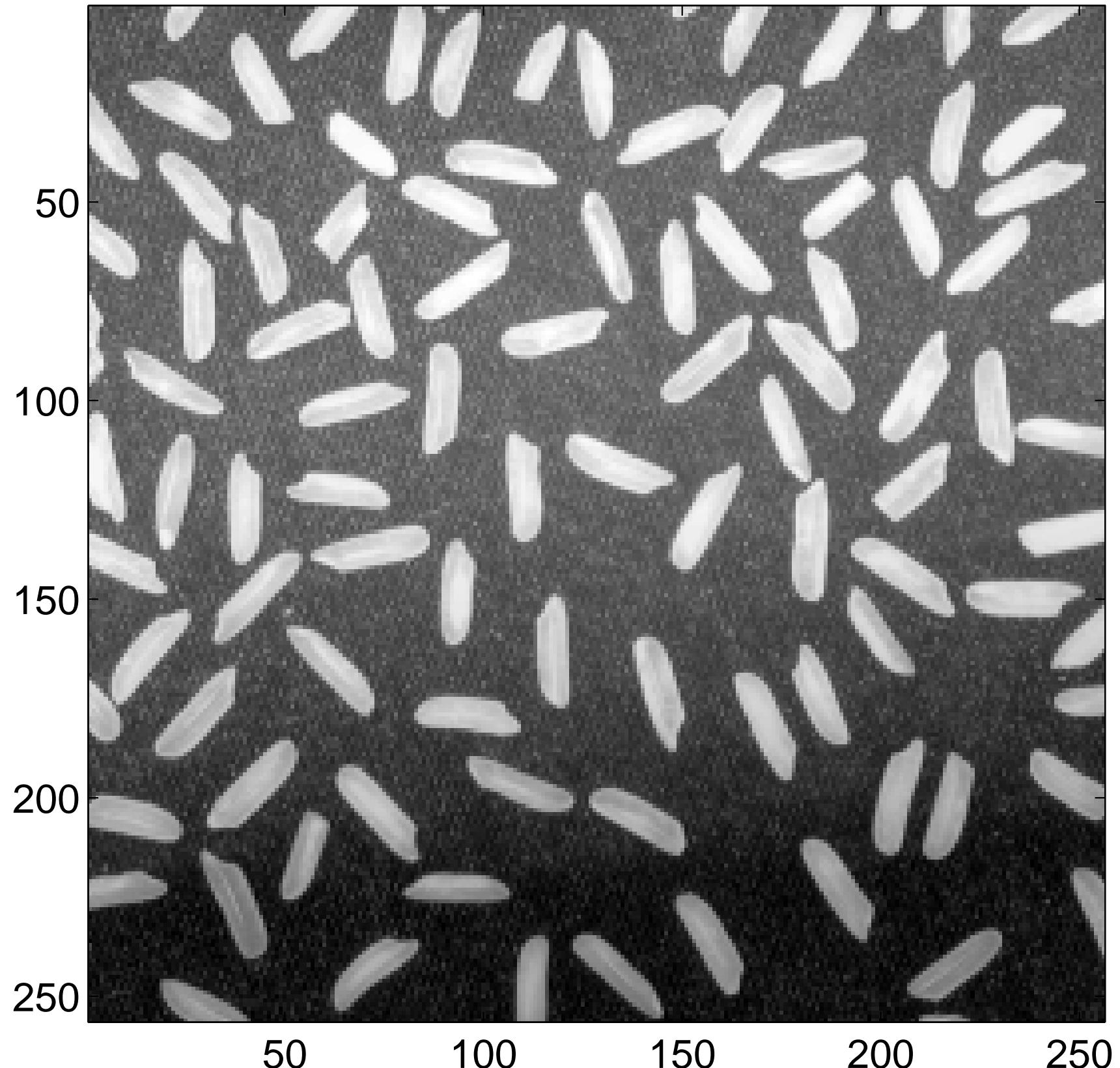


log power spectrum, centered

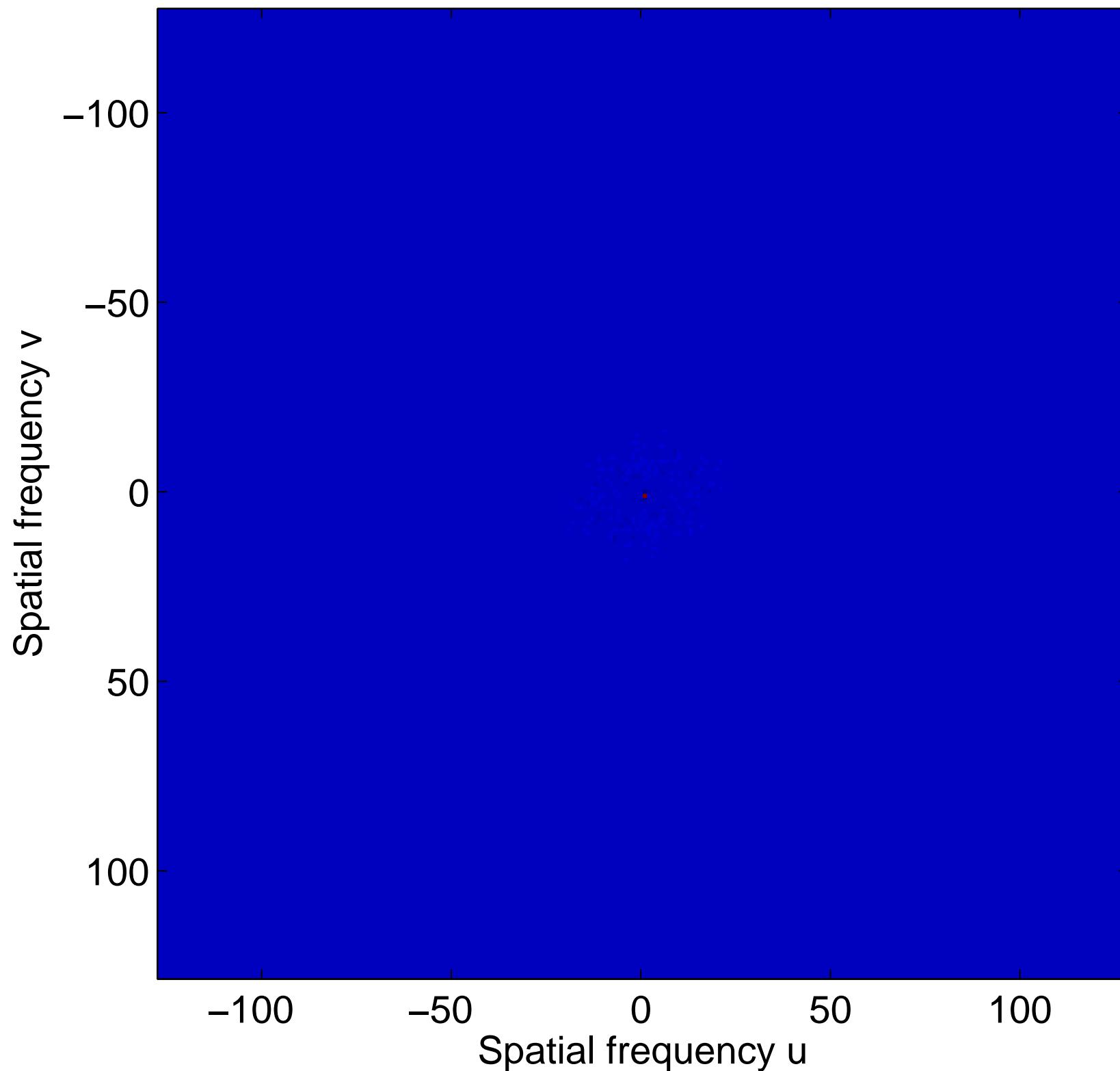


log power spectrum, centered

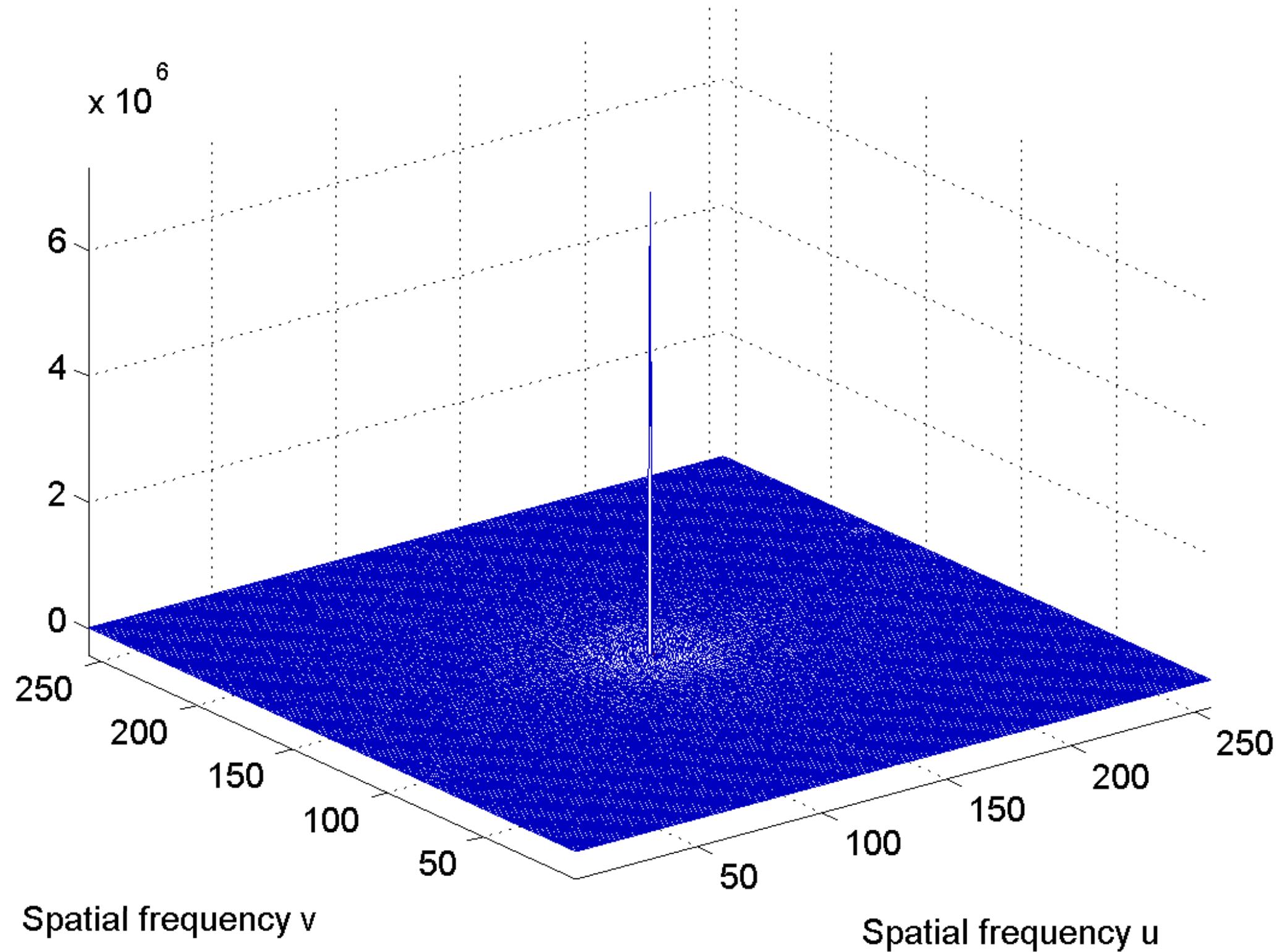




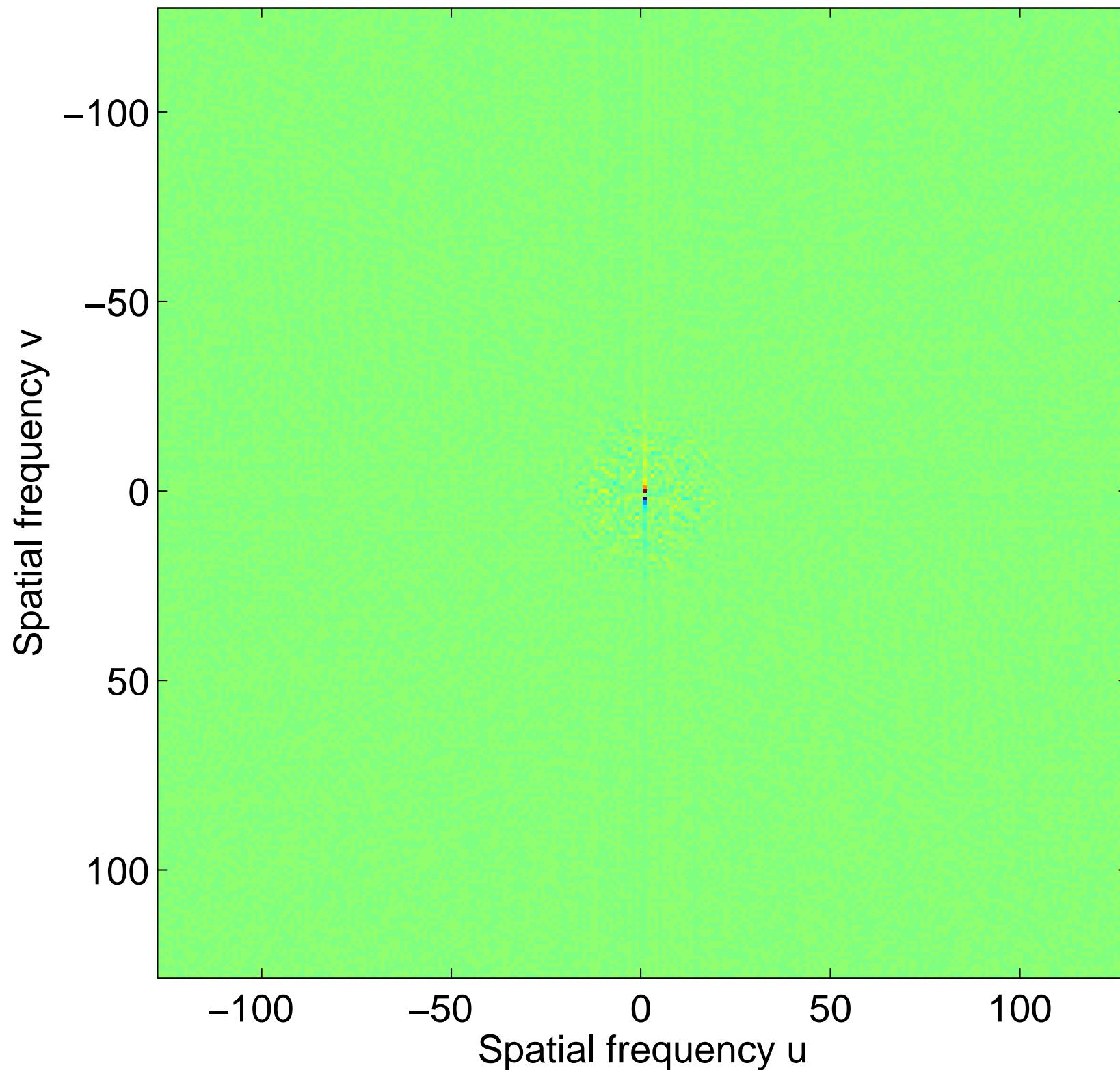
Real part of the spectrum, centered



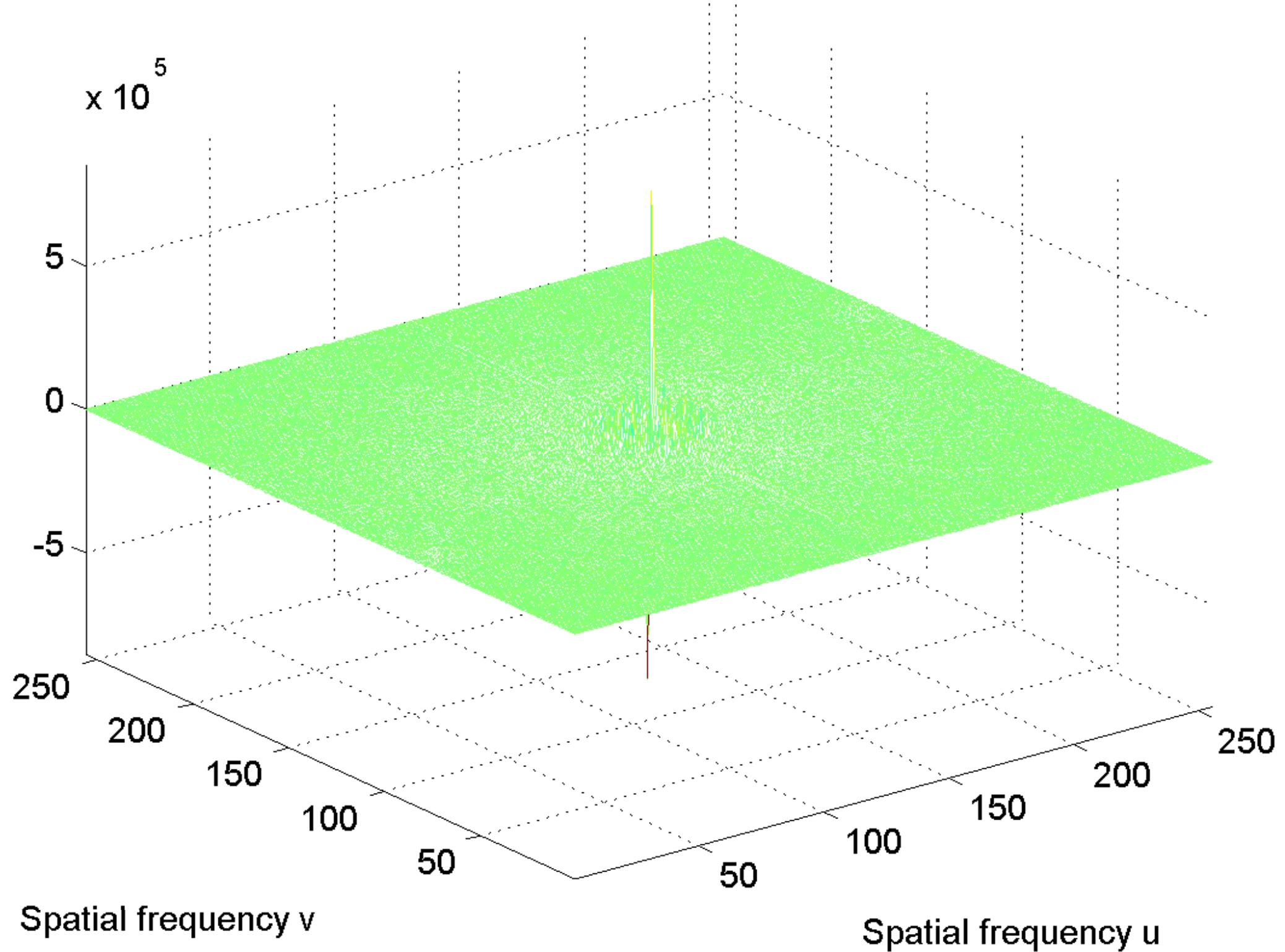
Real part of the spectrum, centered



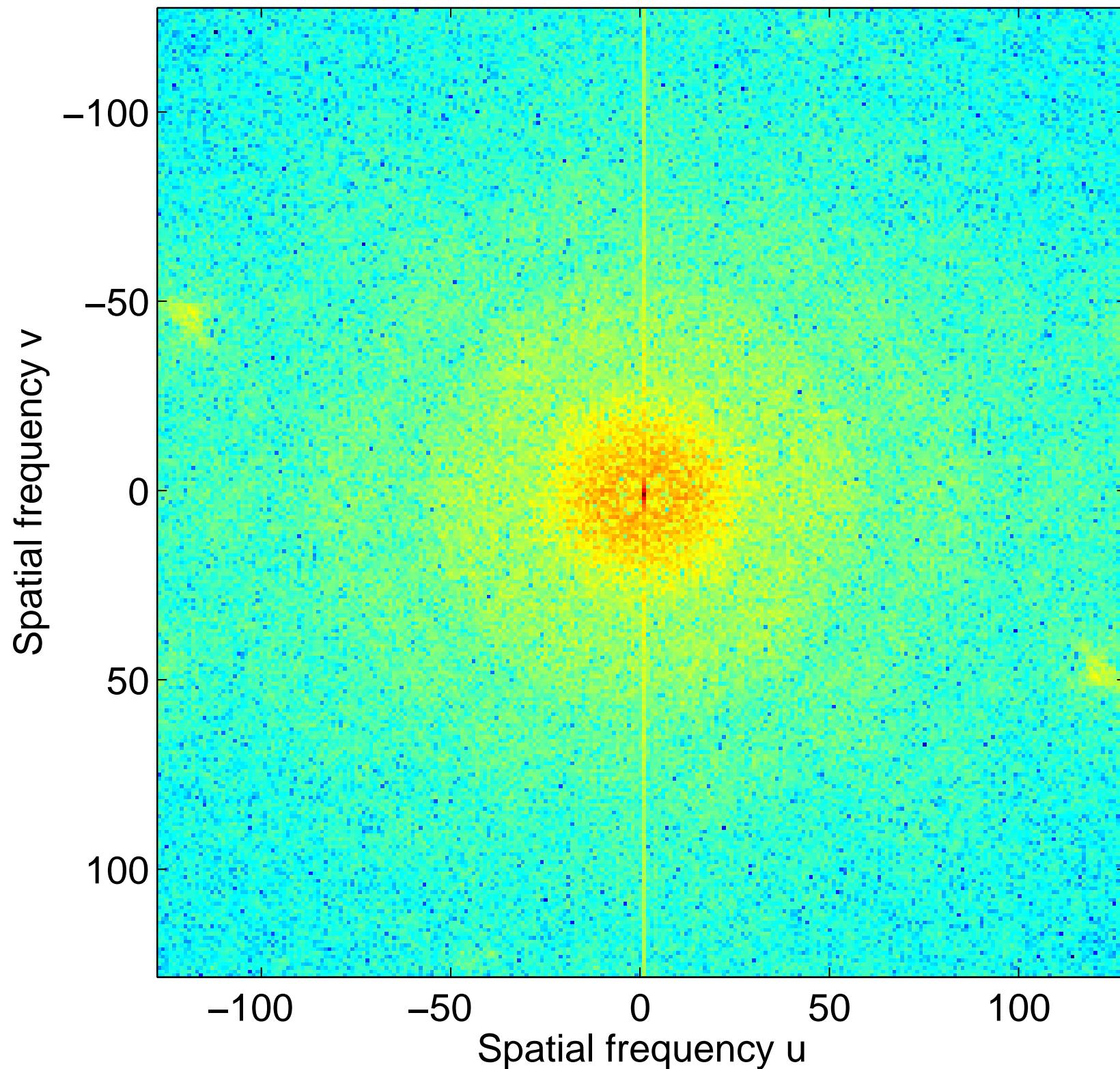
Imaginary part of the spectrum, centered



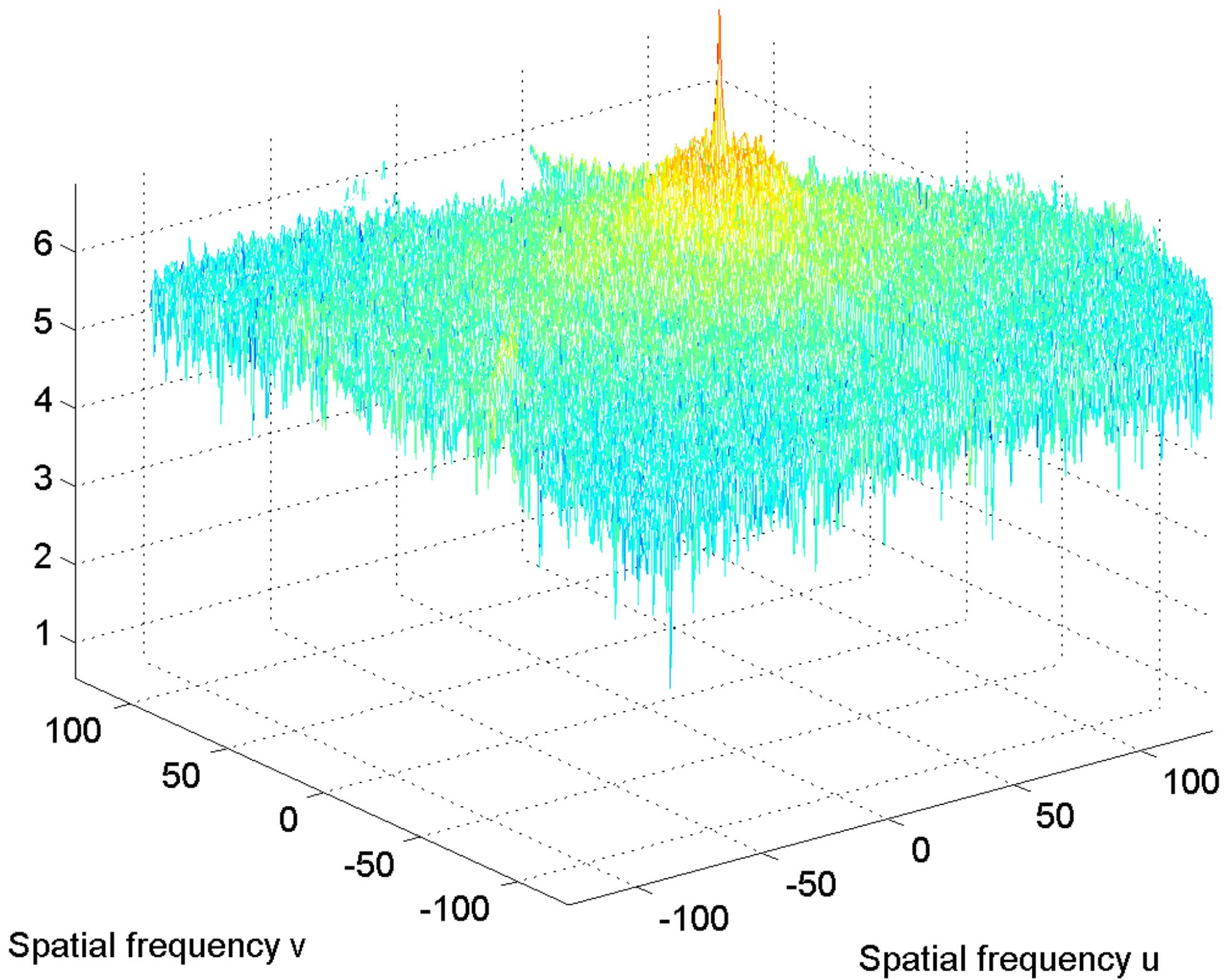
Imaginary part of the spectrum, centered

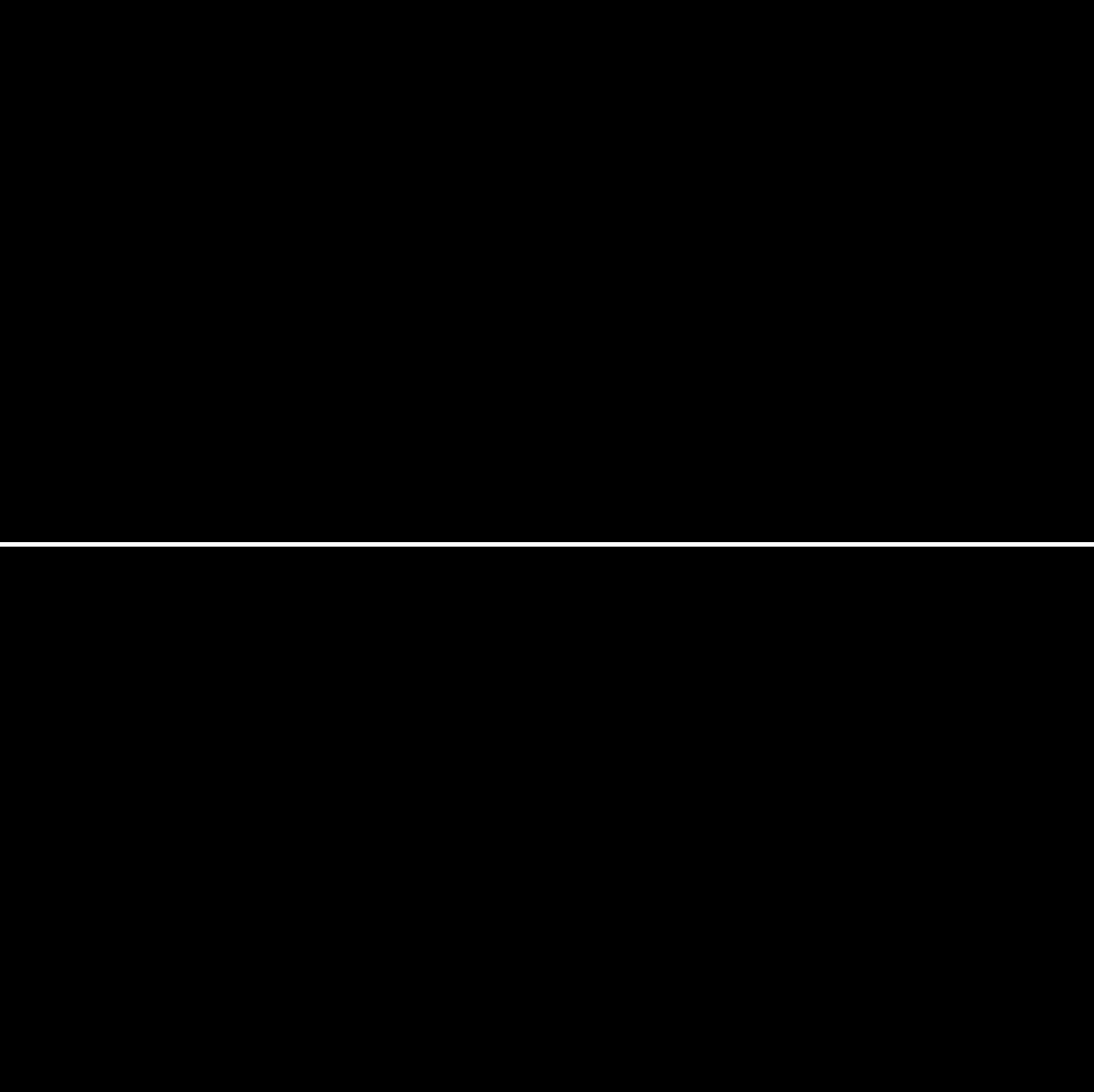


log power spectrum, centered

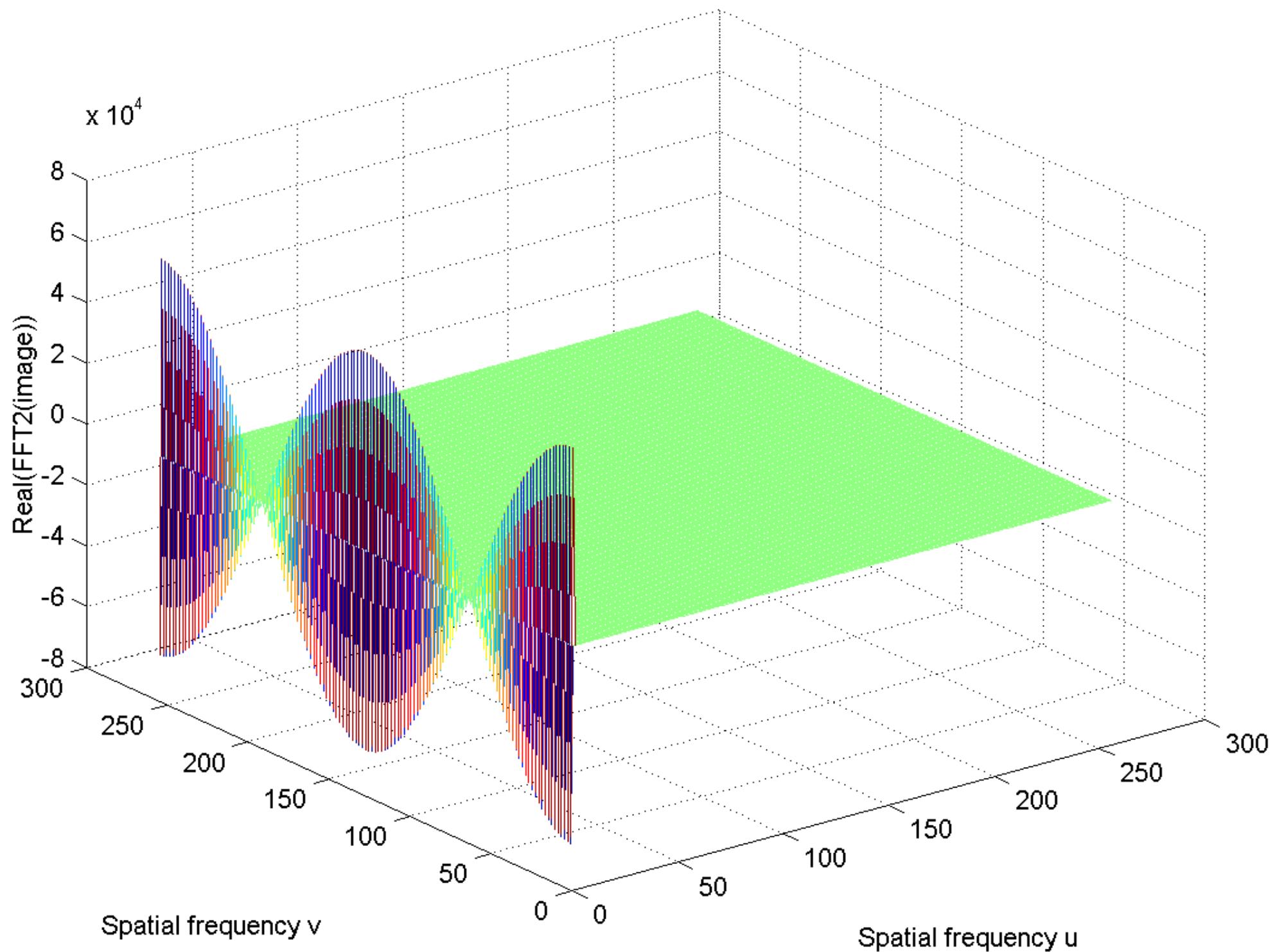


log power spectrum, centered

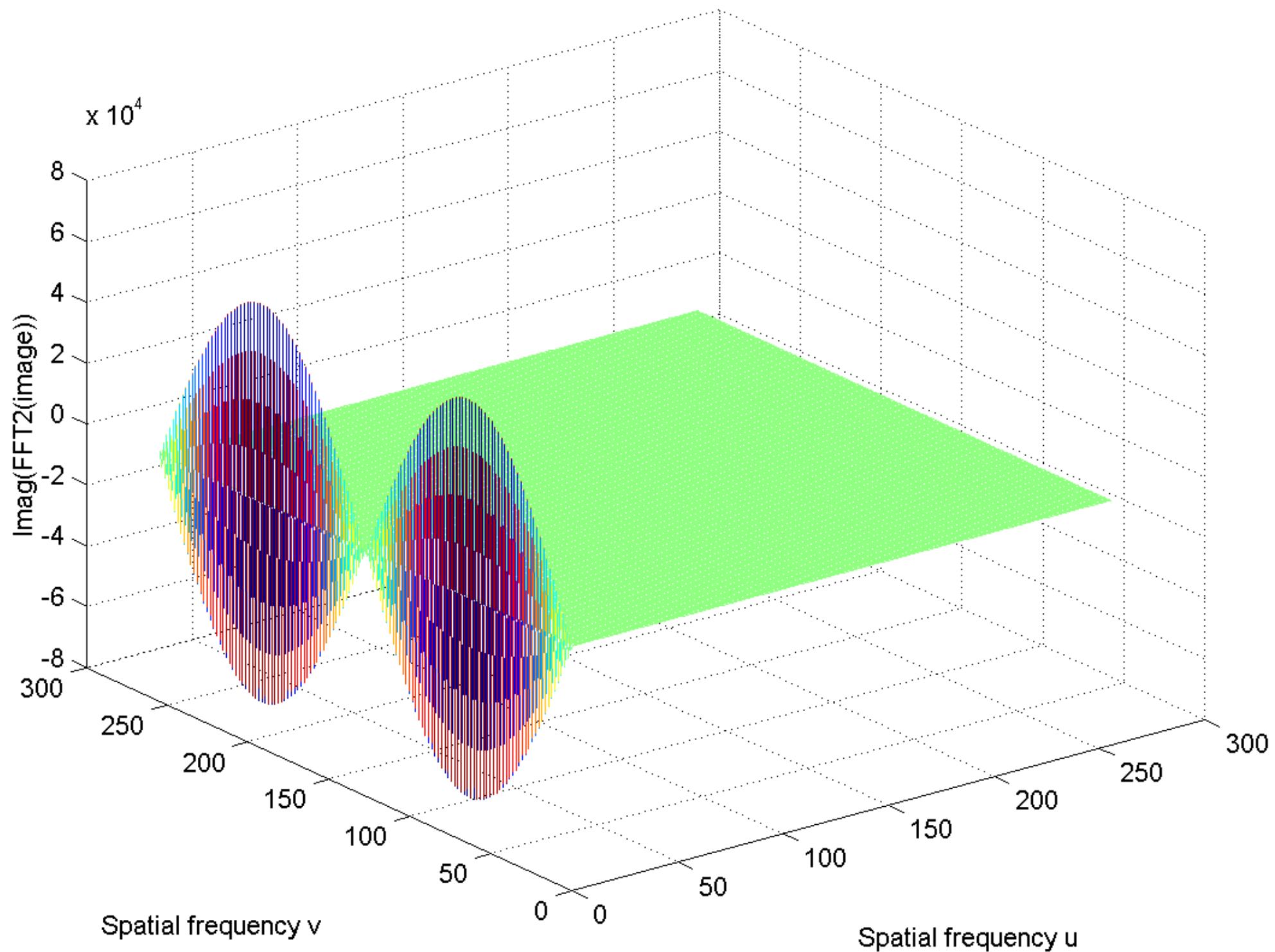




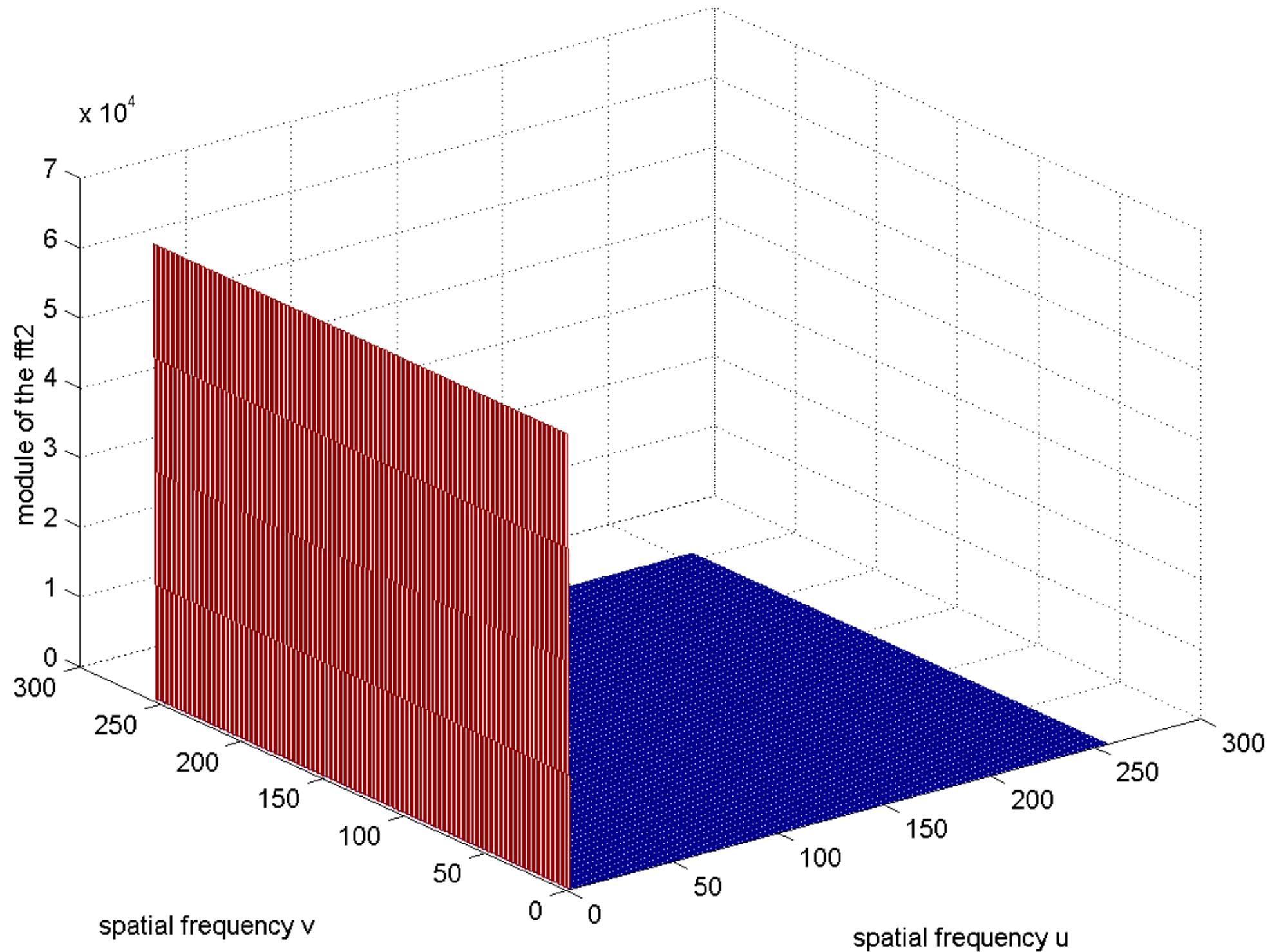
Frequency spectrum, real part of the FFT

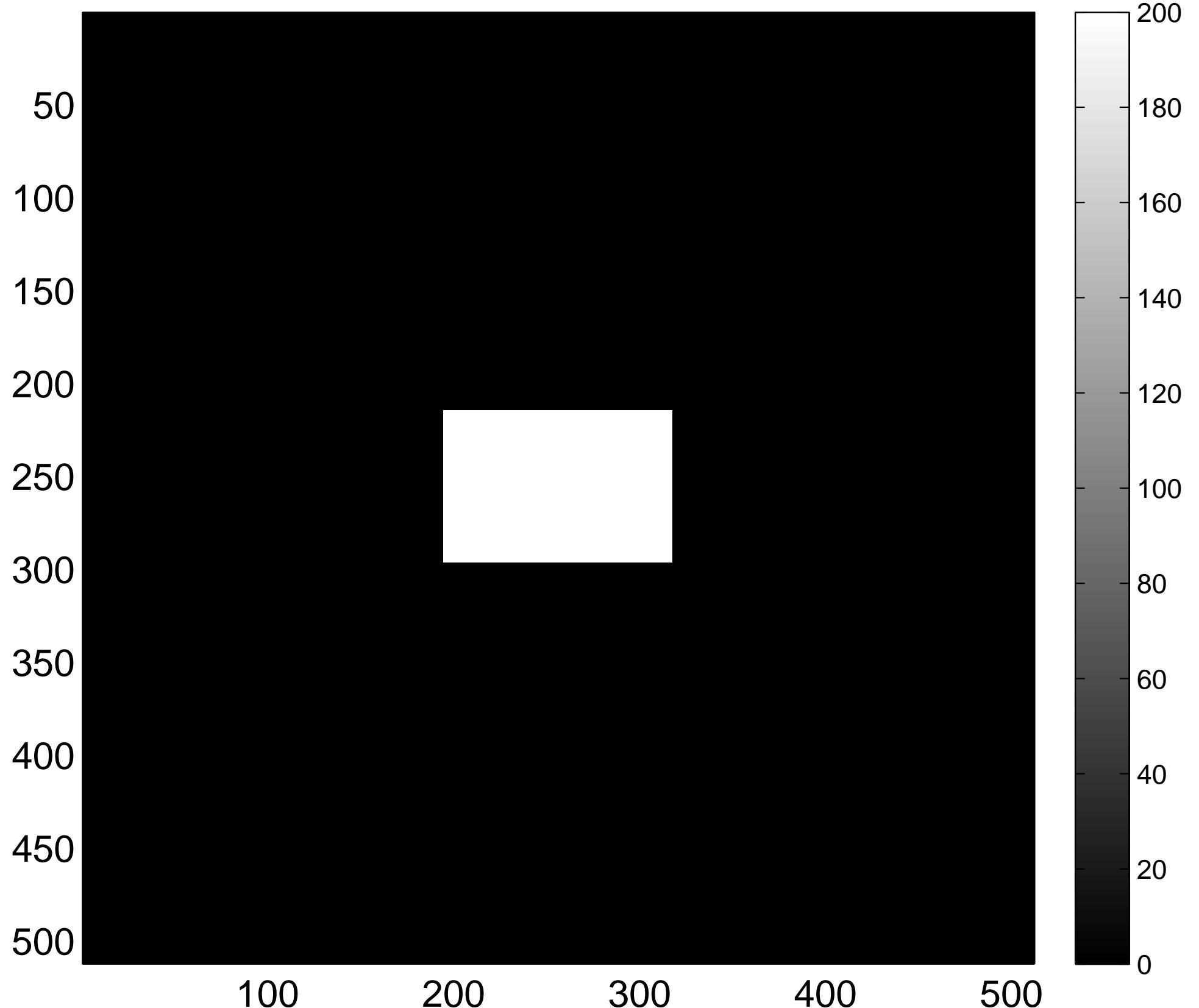


Frequency spectrum, imaginary part of the FFT

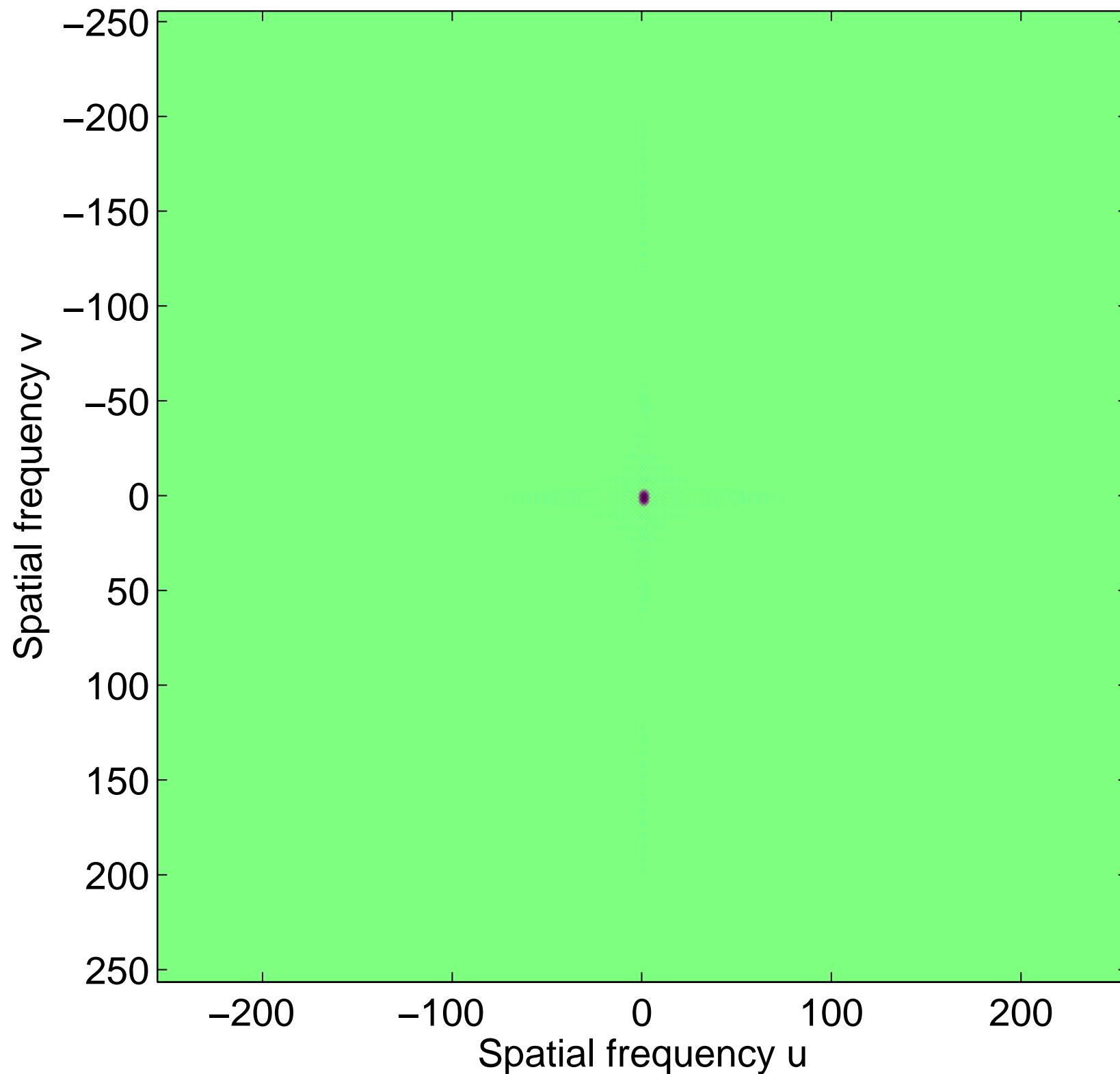


Power spectrum

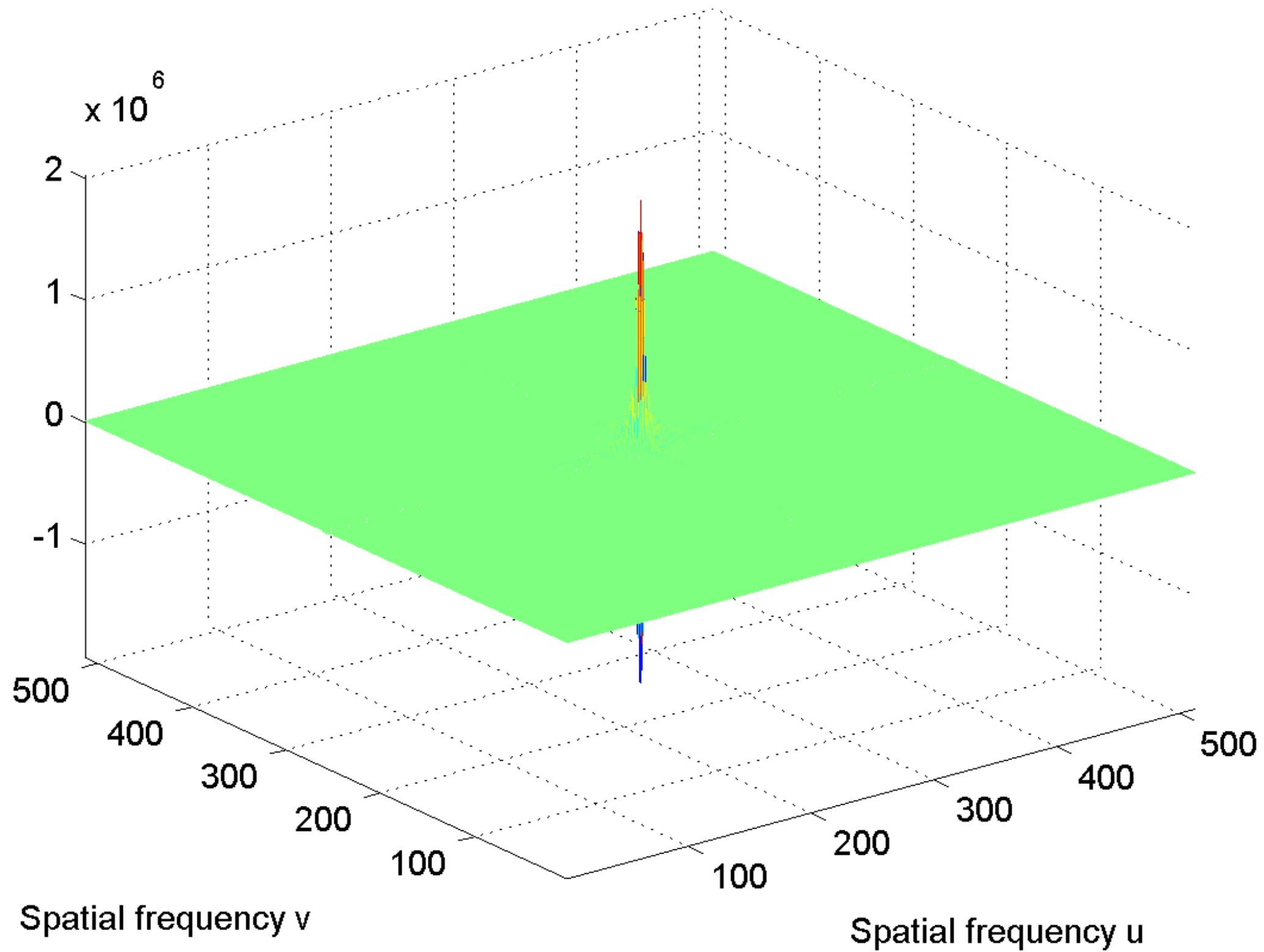




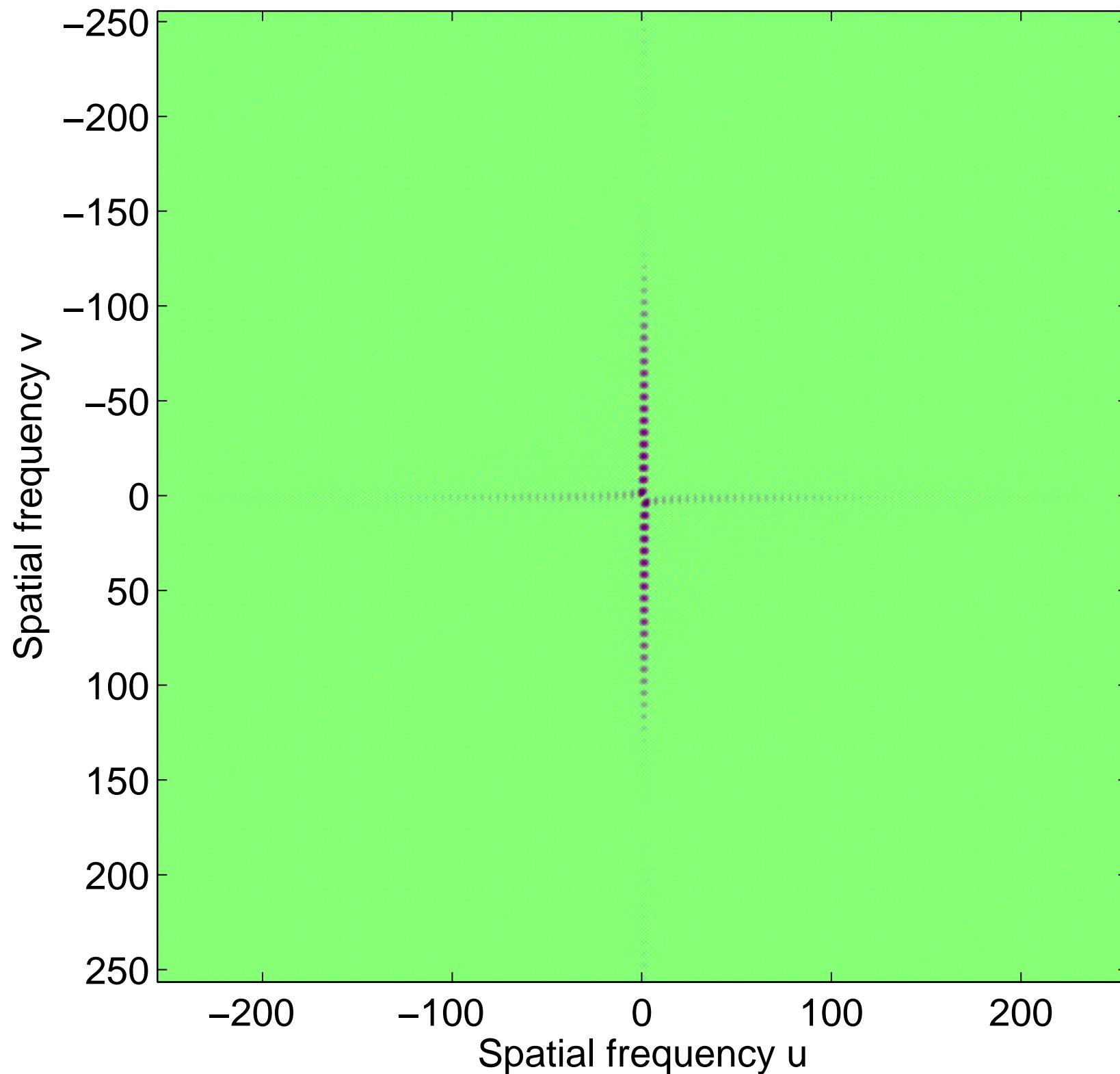
Real part of the spectrum, centered



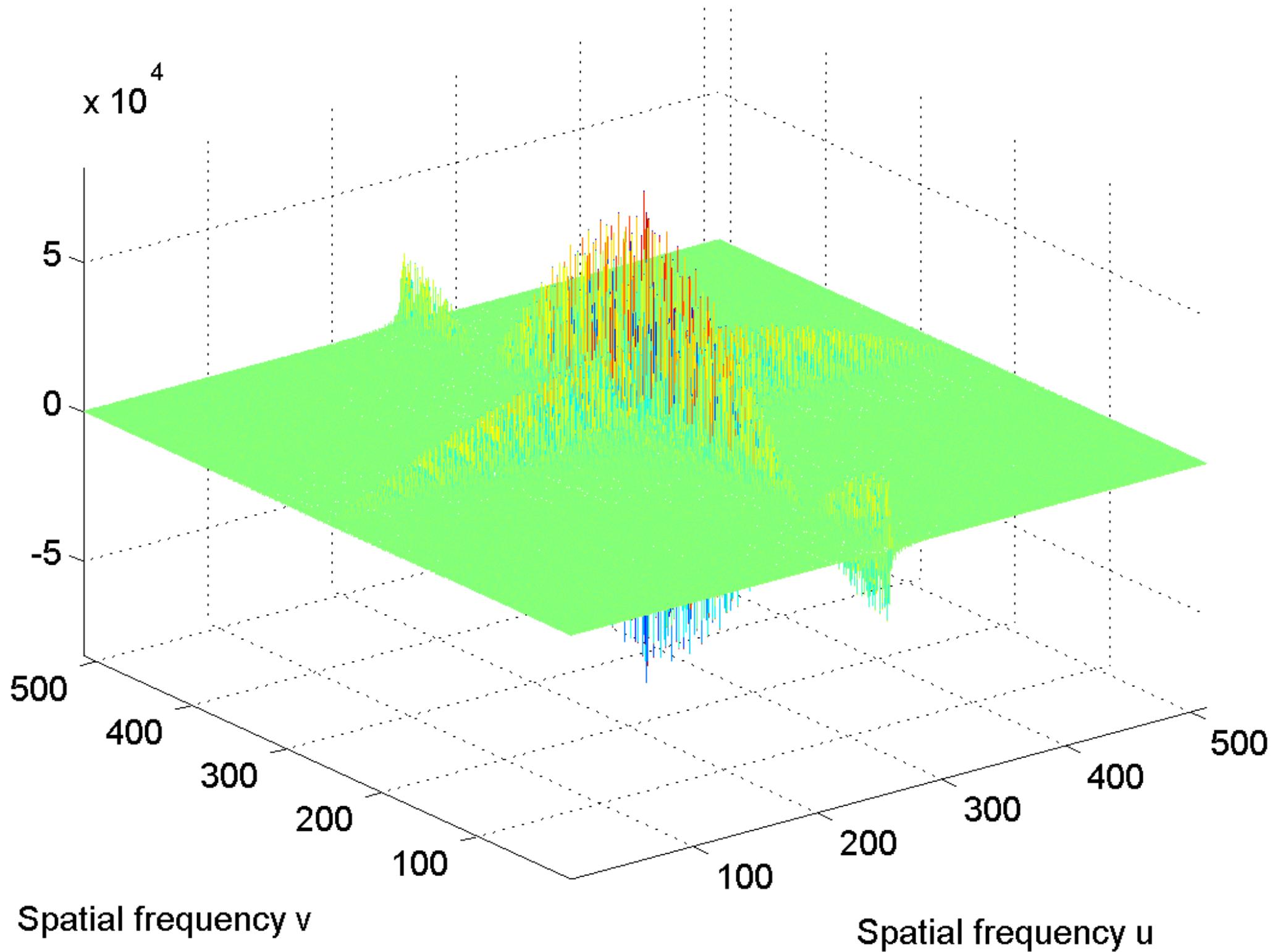
Real part of the spectrum, centered



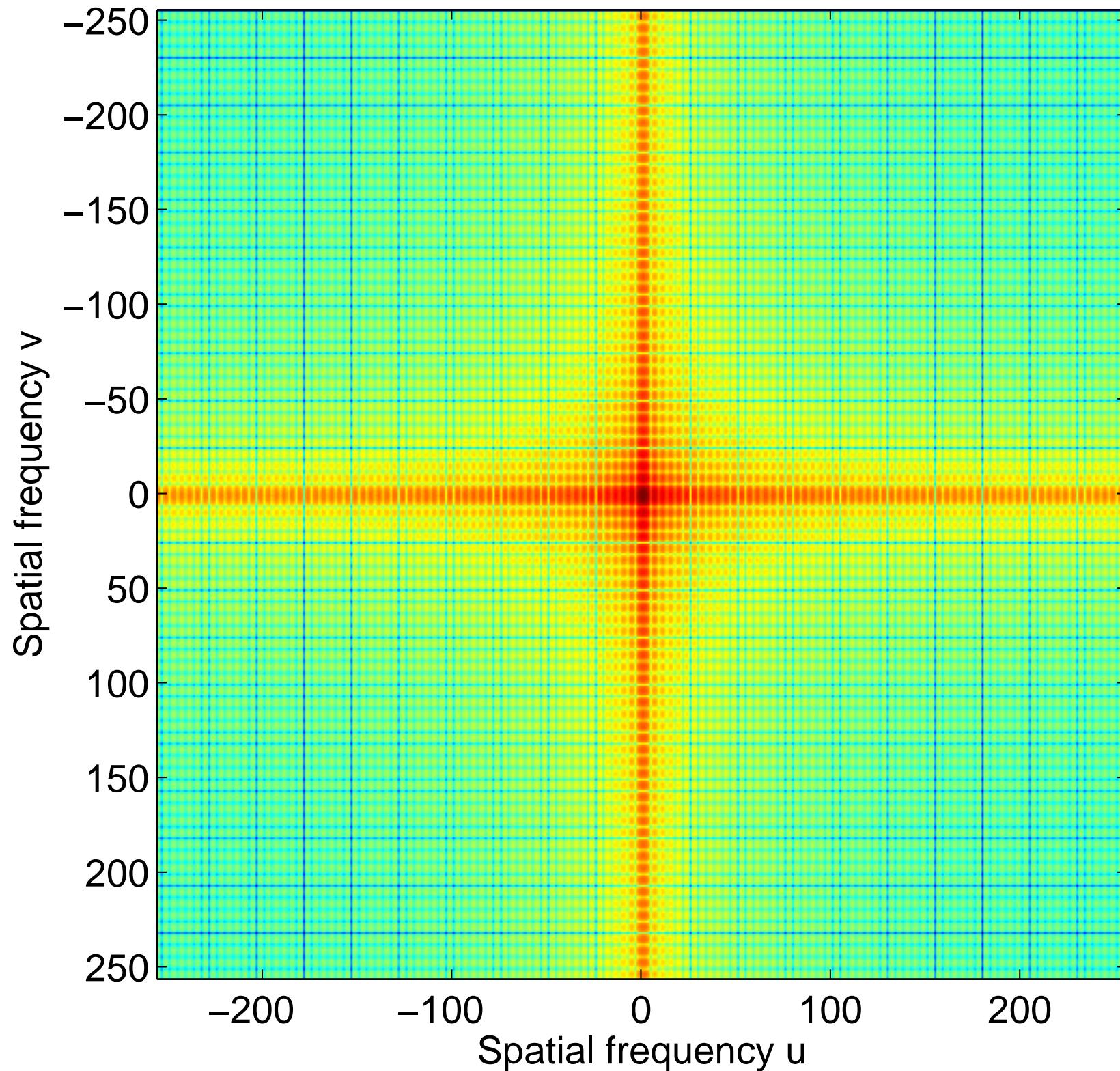
Imaginary part of the spectrum, centered



Imaginary part of the spectrum, centered



log power spectrum, centered



log power spectrum, centered

