Design and Development of Laplacian Pyramid Combined with Bilateral Filtering Based Image Denoising

P. Karthikeyan(✉), S. Vasuki, K. Karthik, and M. Sakthivel

Department of Electronics and Communication Engineering, Velammal College of Engineering and Technology, Madurai, India
kpkarthi2001@gmail.com

Abstract. This paper mainly deals with image denoising algorithm by applying bilateral filter in Laplacian subbands using raspberry pi. Bilateral filter is used for reducing various noises especially Additive White Gaussian Noise which occur more in standard test images. The important feature of this nonlinear bilateral filter is the preservation of the edges, while reducing the noise in the images. The main idea is to replace the pixel's intensity value by a weighted average of adjacent pixel intensity value. Euclidean distance and radiometric differences of pixels, are used for weight calculation. This calculation mainly preserves sharp edges by looping through each pixel and adjusting weights to the nearest pixel values. Our project aims to reduce cost and power consumption using Raspberry pi. It consists of series of microcomputer packed onto a single circuit board. These low power computers are mass produced at very low prices. The performance of Raspberry pi is equivalent to a personal computer. In order to perform noise reduction in Raspberry pi, python2 and opencv package is installed in real-time Raspbian Linux operating system and algorithm is executed using python2 programming language. The denoising method using Laplacian subbands provides better denoised images compared to Gaussian filter and Bilateral filter that applied in standard test images.

Keywords: Image denoising · Laplacian subbands · Gaussian filter Bilateral filter · Raspberry pi

1 Introduction

A digital image is a representation of image using a finite set of digital values called pixels or picture elements. Digital image processing is a domain of signal processing which takes the image as the input, such as a photograph or an video frame; the output may be either an image or a set of characteristics or parameters related to the image.

The image enhancement algorithm aims at reducing noise and also preserves the edges. The images are affected by various noises like Impulse noise, Poisson noise, Additive White Gaussian Noise etc. For example, each pixel in an image with Additive White Gaussian Noise is distributed by Gaussian random variable with zero mean and variance σ^2. Poisson noise occurs in photon counting in optical devices, where it

I. Zelinka et al. (Eds.): ICSCS 2018, CCIS 837, pp. 211–221, 2018.
https://doi.org/10.1007/978-981-13-1936-5_24

resembles with the particle nature of light. The quantization and amplifier noise occurs due to the conversion of electrons to pixel intensity [1].

The quality of overall image depends upon the combination of at least five factors: contrast, blur, noise, artifacts and distortion. While retrieving an image from some sources, it is mainly affected by additive noise, having Gaussian distribution. Image denoising plays a vital role in transmission, formation and display systems of image. Thus, numerous methods have been developed for image denoising. In [2], Bilateral filter smoothes the image by enhancing the edges, using a nonlinear combination of neighboring image intensity values. The intensity values used for combination may be gray or colors based on their photometric similarity and geometric closeness. The combination uses near values than distant values in both domain and range. The slope of the edges are increased to sharpen an image as in the case of adaptive bilateral filter [3].

Image denoising can also be performed using the combination of bilateral filter and wavelet thresholding called as multiresolution bilateral filtering [4]. In [5], Gaussian noise is removed using bilateral filter in natural images. It is implemented using Raspberry pi. In [6], guided image filtering has better behavior near edges to remove and enhance the appearance of an image according to the distance measure between adjacent pixels. It is the fastest edge preserving filter. The Weighted Least Squares (WLS) filter reduces halos by global intensity shifting.

In [7], Gradient Histogram Preservation (GHP) algorithm is proposed to denoise an image. In addition to image denoising, it enhances the texture structures. In [8], a generalized masking algorithm using individual treatment of the model components was proposed. It reduces halo effect using edge preserving filter. In [9], it performs two operation, contour detection and image segmentation.

Wavelet transforms have become a very powerful tool in the area of image denoising. The wavelet coefficients of the image are Wiener filtered to denoise an image degraded by an AWGN [10]. In [11], bilateral filtering in the Laplacian subbands is used to denoise the noisy image which is implemented using Raspberry pi. The pi has a single-core processor that runs at 700 MHz. It has a coprocessor for performing floating point calculations. It has 512 MB of RAM. In pi, the image gets onto a Micro Secure Digital (SD) card instead of a standard-sized SD card.

2 Materials

2.1 Gaussian Filter

Gaussian filters smoothes an image by calculating weighted averages in a filter box. This filter doesn't produce any overshoot to a step function input while minimizing the rise and fall time.

$$\mathrm{GF[I]_a = G_\sigma(\|a - b\|)I(b)} \tag{1}$$

Where

$$G_{\sigma}(d) = \frac{1}{2\pi\sigma^2} e^{-\frac{d^2}{2\sigma^2}} \tag{2}$$

The spatial distance is defined by $G_{\sigma}(\|a - b\|)$ and σ is the standard deviation of the Gaussian distribution.

2.2 Bilateral Filter

Bilateral filter overcomes various problems of Gaussian filter. A bilateral filter is a non-linear filter, which reduces noise without affecting the image edges. Here, the intensity value of each pixel is replaced by a weighted average of intensity values from adjacent pixels. Mathematically, the output of a bilateral filter at a location a is given as follows,

$$BF(I)_a = \frac{1}{W} \sum_{q \in s} G_{\sigma_s}(\|a - b\|) G_{\sigma_r}(|I(a) - I(b)|) I(b) \tag{3}$$

Where $G_{\sigma_s}(\|a - b\|)$ is a geometric closeness function and it is given by

$$G_{\sigma_s}(\|a - b\|) = e^{-\frac{a - b^2}{2\sigma_s^2}} \tag{4}$$

$G_{\sigma_r}(|I(a) - I(b)|)$ is a gray level similarity function and it is defined as,

$$G_{\sigma_r}(|I(a) - I(b)|) = e^{-\frac{|I(a) - I(b)|^2}{2\sigma_r^2}} \tag{5}$$

W is a normalization constant given by,

$$W = \sum\nolimits_{q \in s} G_{\sigma_S}(\|a - b\|) G_{\sigma_r}(|I(a) - I(b)|) \tag{6}$$

$\|a - b\|$ is the Euclidean distance between a and *b,* and *S* is a spatial neighbourhood of *a*.

The parameters σ_s and σ_r describes the bilateral filter behavior. Space: σ_s represent spatial extent of the kernel, size of the considered neighborhood. Range: σ_r minimum amplitude of an edge. The space parameter proportional to image size. The range parameter proportional to edge amplitude.

2.3 Laplacian Pyramid

The original image is filtered using Gaussian filter to produce a low band image. It is done by appropriate kernel bandwidth. The low band image is then removed from the original image to produce a high band image. This process is repeated to produce a set of subband images called Laplacian pyramid [11]. For an image I, the process is described as,

$$G_0 = I, \tag{7}$$

$$G_{k+1} = \downarrow 2, \left(Gaussian\left(G_k\right)\right) \; for \; k = 0, \ldots n-1$$

The Laplacian pyramid is given by,

$$L_{k+1} = G_k - \uparrow 2\left(G_{k+1}\right) \; for \; k = 0, \ldots n-1$$

$$L_{n+1} = G_n \tag{8}$$

2.4 Raspberry Pi

The Raspberry Pi is an ultra cheap minicomputer having length of 9 cm and width of 5.5 cm. The Raspberry Pi has CPU, RAM and GPU in one component called System on Chip (SoC). It uses an ARM1176JZF-S 700 MHz CPU which is single core and also has a co-processor for floating point calculations. The working memory in Raspberry pi is 512 MB SDRAM. The power consumption is low which consumes only 5 to 7 watts of electricity [12] (Fig. 1).

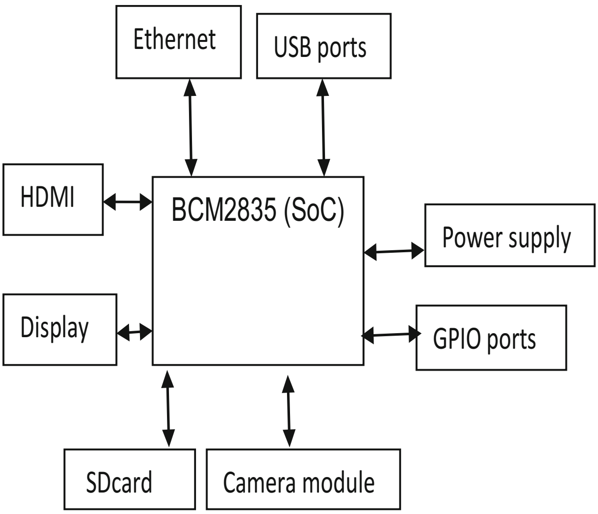

Fig. 1. Raspberry pi hardware architecture

2.5 Memory

The Raspberry Pi processor has two levels of cache memory. Level 1 is smaller in size and has 32 KB of cache memory. Level 2 is large in size and has 128 KB of cache memory. Cache memory has stored the recently used programs. The instructions are executed in the Arithmetic Logic Unit in CPU. CPU produces accurate integer calculation results (Table 1).

Table 1. Raspberry Pi specification

Chip	Broadcom BCM2835 SoC
Core architecture	ARM11
CPU	700 MHz Low Power ARM1176JZFS Applications Processor
Memory	512 MB SDRAM
Operating System	Boots from SD card, running a version of the Linux operating system
Dimensions	85.6 x 53.98 x 17 mm
Power	Micro USB socket 5 V, 1.2A

2.6 Requirements

The Raspberry pi is a tiny, versatile device and also integrates inside of other devices too. It should have both hardware and software requirements.

- The Raspberry pi need an SD card and power supply for connecting keyboard and mouse. It should have a display for running operating system like Linux, Microsoft Windows. The display
- The software requires a cross compiler that converts source code files into Raspberry Pi-compatible executable files which is in the SD card. The word length is 32 bit. A word is defined as a complete information that the CPU can execute. These instructions are executed by Arithmetic Logic Unit (ALU) which is an important part of the CPU.

2.7 Networking

The model A and A+ can be connected to a system by an external user supplied USB Ethernet or an wifi adapter due to lack of an 8P8C ("RJ45") USB port in Ethernet. In case of model B and B+ the USB port is having build-in Ethernet adapter.

2.8 Power Supply

The raspberry pi is powered by the supply voltage of 5v DC power supply. The power supply used by model B is 700 mA to 1200 mA. When no peripherals are attached model A uses 500 mA power supply. The raspberry pi mainly uses below 1A. The processor becomes unstable if the supply voltage is below 4.75v. The GPIO pins requires power of 50 mA and keyboard, mice requires below 100 mA or over 1000 mA.

2.9 Essential Peripherals

- High Definition Multimedia Interface (HDMI) by the use of a single cable it supports high-quality digital video and audio [13].
- Composite video do not support HDMI which mainly provides the Raspberry Pi to be connected to a Television. It does not produce output in high quality as HDMI and is an analogue standard.
- By using an converter, it is also possible to connect a computer monitor with a Digital Visual Interface (DVI) connection to the HDMI. The two of these standards are digital.

2.10 Keyboard and Mouse

The Raspberry Pi uses the USB Keyboard and Mouse by plugging into it also if paired wireless keyboard and mouse can also be used. Midi sized keyboards can also be integrated into the track pads. It requires less space than the standard keyboards (Fig. 2).

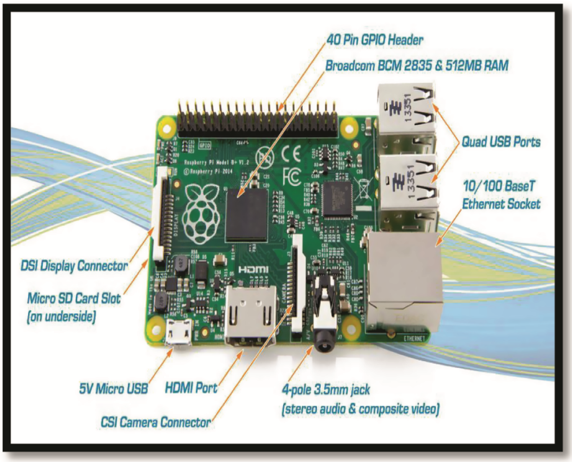

Fig. 2. Raspberry pi B + module

3 Methods

3.1 Denoising Algorithm

- Let the image be considered as I. The image I is affected as AWGN having variance σ that mainly occurs in natural images.
- The noisy image is filtered using Gaussian filter by downsampling to obtain a low band image $G_{k+1} = \downarrow 2G_K$.
- The Laplacian subbands are obtained from $L_{K+1} = G_K - \uparrow 2\left(G_{K+1}\right)$.
- The process gets a set of subband signals called Laplacian pyramid.

- Applying bilateral filter in the Laplacian pyramid for denoising.
- Evaluate the Peak Signal to Noise Ratio and Image Quality Index values in Raspberry pi using Python2 programming.
- Compare the Peak Signal to Noise Ratio for Gaussian filter, bilateral filter in natural images and the combination of Laplacian pyramid and bilateral filter.

The performance of noise reduction at the slope and slowly varying areas which become flat in the high band. The Laplacian sub-band bilateral filtering can be a considerable choice for denoising the noisy images mainly for fast implementation of real-time systems. The proposed method improves Peak Signal to Noise Ratio of the image compared to the bilateral filtering and subband decomposition methods.

The subband bilateral filtering algorithm is compared with the original bilateral filtering $\sigma_{d\,=}$ 1.8 and $\sigma_{r\,=}$ 2σ. The images are corrupted by using 100 different noisy sequences of Gaussian distribution with variance of 20 to 50. The Peak Signal to Noise

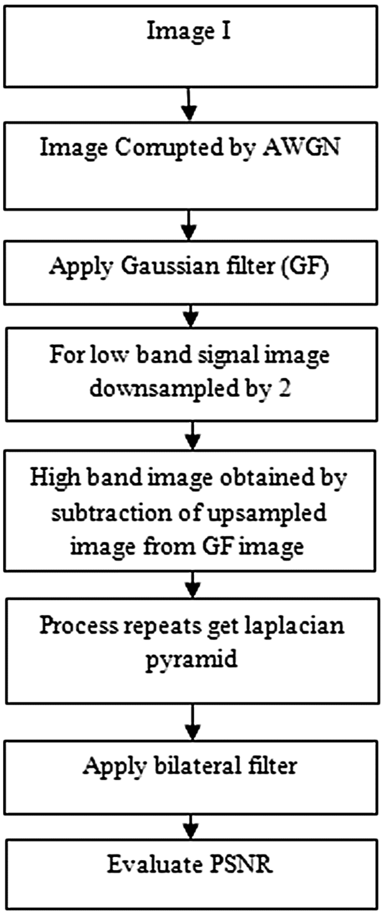

Fig. 3. Work flow of proposed denoising algorithm

Ratio is calculated and compared with the original bilateral filtering by python2 programming language using Raspberry pi [11] (Fig. 3).

4 Results and Discussion

Experiments were performed in images having Additive White Gaussian noise of zero mean and variance values σ_n varies from 20 to 50. The images are denoised by applying bilateral filter in the Laplacian subbands. The peak signal to noise ratio and image quality index values of the proposed method are calculated and compared with Gaussian and Bilateral filter. The proposed method produces high peak signal to noise ratio and Image Quality Index value compared to Gaussian and bilateral filter applied in standard test images.

The peak signal to noise ratio are calculated from the mean squared error value between the original image and compressed image.

The high quality denoised images produces high peak signal to noise ratio. The peak signal to noise ratio calculation in terms of mean squared error is given by

$$\text{PSNR(dB)} = 10\log_{10}\frac{255^2}{\text{MSE}} \tag{11}$$

where, mean squared error is given by,

$$\text{MSE} = \frac{1}{\text{NXN}}\sum\nolimits_{i=0}^{N-1}\sum\nolimits_{j=0}^{N-1}\left|x_{i,j} - \hat{x}_{i,j}\right|^2 \tag{12}$$

The personal computer or laptop is very costly. It also consumes large amount of power, as well as are very costly. This paper mainly involves replacement of computer with an ultra-low cost and low power consuming device called Raspberry pi. It is a hand held device and it is used for teaching in schools, colleges and also for presentation in seminars. Figure 4 shows the Gaussian Filter output in python2. Figure 5 shows the bilateral filter output and Fig. 6 shows the subband bilateral filter output in python2 (Fig. 7, Table 2).

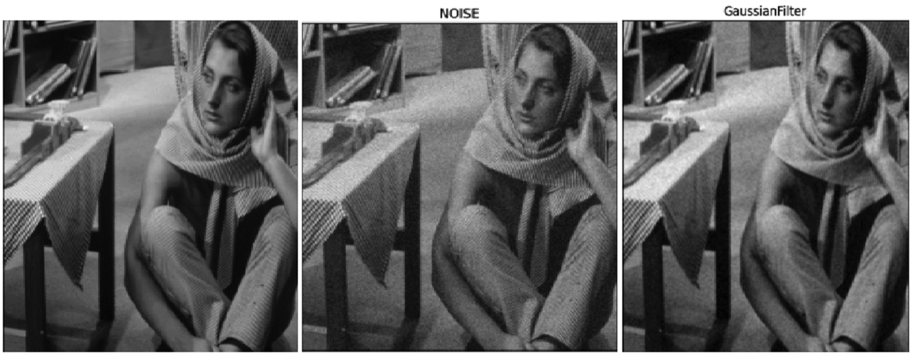

Fig. 4. Gaussian filter output of Barbara image in python2

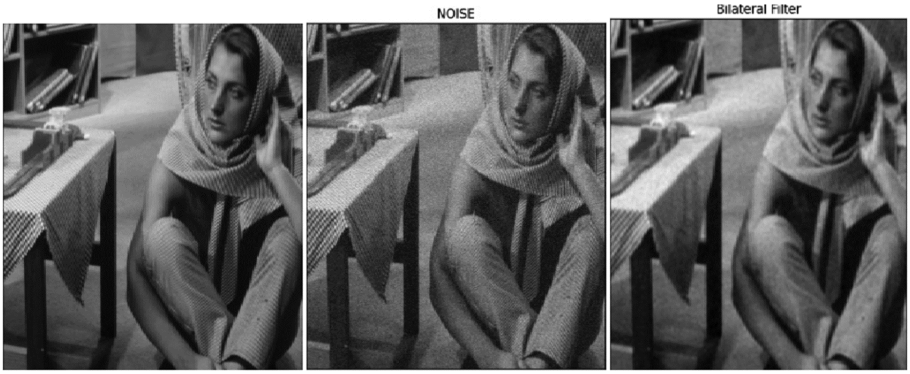

Fig. 5. Bilateral filter output of Barbara image in python2

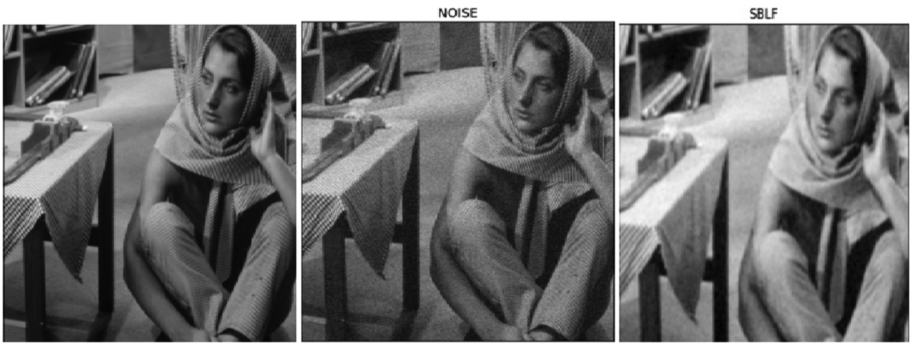

Fig. 6. Laplacian pyramid + BLF output of Barbara image in python2

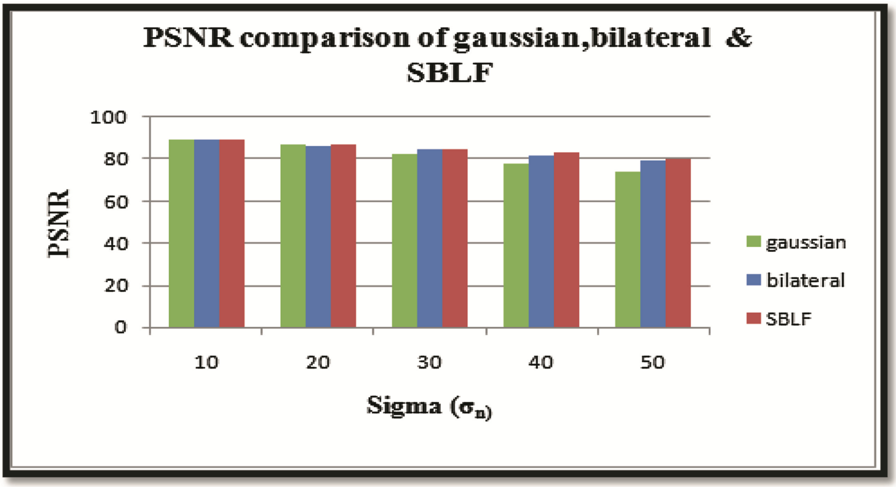

Fig. 7. Peak Signal to Noise Ratio comparison of Gaussian, bilateral & SBLF

Table 2. Peak Signal to Noise Ratio comparison between various denoising methods and proposed method

Test images	σ_n	Gaussian	Bilateral	Laplacian pyramid + bilateral
Barbara	10	89.5272	89.1661	89.6782
	20	86.9389	86.5595	87.1245
	30	82.1330	84.5762	85.0178
	40	77.5564	81.9093	83.1002
	50	73.6793	79.0986	80.1671
Lena	10	90.4026	90.9518	90.9743
	20	86.1894	86.5475	86.7864
	30	80.3010	83.8302	84.1456
	40	77.5714	83.3651	83.9543
	50	72.4569	80.1578	80.6752
Cameraman	10	82.3680	86.8563	87.2134
	20	77.1461	86.0642	86.9761
	30	75.8910	80.8823	81.7815
	40	75.3506	78.7431	79.4537
	50	70.4389	74.0023	75.0238

5 Conclusion

Raspbian wheezy OS was installed successfully on SD card in Raspberry Pi Kit. To suppress the Additive White Gaussian Noise python2 program was developed. The combination of Laplacian pyramid and bilateral filter method produce improved results than the filters applied in standard test images. The proposed method gives outperformed results than Gaussian and bilateral filter in terms of Peak Signal to Noise Ratio and Image Quality Index values. Using this method it produces 25% increase in Peak Signal to Noise Ratio and 10% increase in Image Quality Index value. Personal computer consumes large amount of power and also expensive. Thus, this project mainly use low cost Raspberry pi instead of personal computer. In addition to low cost, CPU usage of 7.3% can be designed using Raspberry pi. The proposed framework will motivate further research towards better understanding of raspberry pi.

References

1. Gonzalez, R.C., Woods, R.E.: Digital Image Process. Pearson Education (Singapore) Pte. Ltd., Delhi (2004)
2. Tomasi, C., Manduchi, R.: Bilateral filtering for gray and color images. In: Proceedings of the IEEE International Conference on Computer Vision, Bombay, pp. 839–846 (1998)
3. Zhang, B., Allebach, J.P.: Adaptive bilateral filter for sharpness enhancement and noise removal. IEEE Trans. Image Process. **17**(5), 664–678 (2008)
4. Zhang, M., Gunturk, B.K.: Multiresolution bilateral filtering for image denoising. IEEE Trans. Image Process. **17**(12), 2324–2333 (2008)

5. Karthikeyan, P., Vasuki, S., Boomadevi, R.: Effective noise removal in graylevel image using joint bilateral filter. Int. J. Appl. Eng. Res. **9**(21), 4831–4836 (2014). ISSN 0973-4562
6. He, K., Sun, J., Tang, X.: Guided image filtering. IEEE Trans. Pattern Anal. Mach. Intell. **35**(6), 1397–1409 (2013)
7. Zuo, W., Zhang, L., Song, C., Zhang, D.: Texture enhanced image denoising via gradient histogram preservation. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 1203–1210 (2013)
8. Deng, G.: A generalized unsharp masking algorithm. IEEE Trans. Image Process. **20**(5), 1249–1261 (2011)
9. Arbelaez, P., Maire, M., Fowlkes, C., Malik, J.: Contour detection and hierarchical image segmentation. IEEE Trans. Pattern Anal. Mach. Intell. **33**(5), 898–916 (2011)
10. Jacob, N., Martin, A.: Image Denoising in the Wavelet Domain Using Wiener Filtering, December 2004
11. Jin, B., You, S.J., Cho, N.I.: Bilateral image denoising in the Laplacian subbands. EURASIP J. Image Video Process. **2015**(1), 1–12 (2015)
12. https://www.slideshare.net/anija03/Raspberry_Pi
13. Quick Start Guide, the Raspberry Pi – Single Board Computer. https://www.farnell.com/datasheets/1524403.pdf