



An efficient and adaptable multimedia system for converting PAL to VGA in real-time video processing

Deepak Kumar Jain¹ · Sunil Jacob² · Jafar Alzubi³ · Varun Menon⁴

Received: 2 February 2019 / Accepted: 4 June 2019 / Published online: 12 June 2019
© Springer-Verlag GmbH Germany, part of Springer Nature 2019

Abstract

Real-time video processing has found its range of applications from defense to consumer electronics for surveillance, video conferencing, etc. With the advent of Field Programmable Gate Arrays (FPGAs), flexible real-time video processing systems which can meet hard real-time constraints are easily realized with short development time. Most of the existing solutions have high utilization of system resources and are not quite flexible with many applications. Here we propose a hardware–software co-design for an FPGA-based real-time video processing system to convert video in standard Phase Alternating Line (PAL) 576i format to standard video of Video Graphics Array (VGA)/Super Video Graphics Array (SVGA) format with little utilization of resources. Switching between multiple video streams, character/text overlaying, and skin color detection are also incorporated with the system. The system is also adaptable for rugged applications. VHSIC Hardware Description Language (VHDL) codes for the architecture were synthesized using Altera Quartus II and targeted for Altera Stratix I FPGA. Results achieved confirm that the proposed system performs efficient conversion with very less resource utilization compared to the existing solutions. Since the proposed system is also flexible, many other applications can be incorporated in the future.

Keywords FPGA · Hardware–software co-design · PAL · RTVPS · VGA/SVGA · VHDL

1 Introduction

Nowadays digital information in the form of digital audio, images, and video has become an inevitable part of our daily lives. Real-time digital video has found numerous applications in various fields like video conferencing, surveillance, medical imaging, remote vehicle guidance systems, multimedia-based consumer electronics products, security systems, manufacturing, etc. Real-time video processing is a challenging task that requires substantial computations for

huge data represented by images. Along with functionality, the real-time video processing system (RTVPS) must also be rugged enough to operate in all environmental conditions. Special attention should also be given in the hardware design to meet these requirements.

Real-time image processing involves converting and altering of images immediately or in real time, avoiding any delay experienced by users [1]. Whether the RTVPS does video filter or encode/decode, image alteration or image compression/decompression or any other video processing methods, the main focus is to finish the acquisition of the video signal, its execution, storage, and display. Also the lag from input to output time should be very less to meet the acceptable time limits. It is very important to note that the precision of the computation depends on the results as well as on the time at which the output is generated [2].

In real-time video processing, hardware implementations are considered faster and the better choice due to numerous advantages [3–6]. Recently, most of the researches have focused on implementing efficient hardware solutions. Features like parallel processing architectures, embedded hardware multipliers, huge memory blocks, extremely high memory bandwidth enable video applications with Field

✉ Deepak Kumar Jain
deepak@cqupt.edu.cn

¹ Key Laboratory of Intelligent Air-Ground Cooperative Control for Universities in Chongqing, College of Automation, Chongqing University of Posts and Telecommunications, Chongqing 400065, China

² Center for Robotics, SCMS School of Engineering and Technology, Kerala 683576, India

³ School of Engineering, Al-Balqa Applied University, Salt 19117, Jordan

⁴ Department of Computer Science and Engineering, SCMS School of Engineering and Technology, Kerala 683576, India

Programmable Gate Arrays (FPGAs) to outperform conventional Digital Signal Processors (DSPs) [7]. They offer accurate execution times and help to meet hard real-time deadlines. Interface to multiple external devices can be easily set up with FPGA. As the devices are reprogrammable, they can be easily reconfigured to a totally distinct circuit. In RTVPS, a group of functions are recurrently executed on each frame of the image in the video stream. These functions are highly rigorous and also the precision of these applications are tremendously affected by their accuracy. Most of the important logic resources necessary in RTVPS operation are presently ready optimized and fixed in latest FPGAs [9].

A video processing system with VGA interface is presented in [10]. Using the DSP + FPGA + SCM + ASIC architecture, it finishes the procurement of the video signal and display. This technique also focuses on major technologies of auto focusing and auto iris in FPGA. Difficulty with this design is that its hardware dependency is more and the use of the exclusive chip would introduce multiple complications such as high complexity, increased cost, and bulkiness. While in the proposed design, FPGA is configured to perform all those functionalities achieved by different controllers in the previous design. In [2], a reasonable hardware and software design for the realization of the functions based on research of existing digital image processing algorithms is discussed. On this basis, an image processing system with FPGA is proposed and presented with the details of image procurement, storage, execution, and display. Difficulty with this design is that the entire pre-processing part is accomplished by the custom hardware. Even though design development is faster, flexibility of the design is reduced.

Hardware Description Language (HDL) is best suited for easy development of RTVPS where hardware platform of interest is FPGA. FPGAs allow application-specific custom circuits to be designed using HDL. For C programming language, rapid development of algorithm is difficult. Generally, there are multiple problems with usage of C language data types in real time. In numerous occasions, floating points have to be converted to corresponding integer version. A RTVP system integrating a FPGA chip and a DSP processor was presented in [12]. In the presented method, the FPGA chip is employed as the major module for sampling the image and the processor is utilized for processing the image. An FPGA-based system using HDL will have much more flexibility. Altera Stratix I FPGA is used because it has extensive triggers and I/O pins, shorter design cycle, low power dissipation, and is adaptable with complementary metal-oxide semiconductor (CMOS) or transistor–transistor logic (TTL) level [11]. Industrial grade FPGA of this family is selected for use in our system to make the system adaptable for rugged applications. In obtaining high-speed data, FPGA has incomparable merits to single-chip micro-computer and DSP.

With the above discussed advantages, we propose an efficient hardware–software co-design for an FPGA-based real-time video processing system to convert video in standard PAL 576i format to standard video of VGA/SVGA format with little resource utilization. This article is organized into five sections. The system overview is presented in Sect. 2. Section 3 explains the hardware–software co-design implemented in the system. This section explains the architecture, implementation, and working of the proposed video capture, video control, function expansion, and video display modules. Further, switching between multiple video streams, character/text overlaying, and skin color detection is explained in this section. The results from implementation are discussed in Sect. 4 and the paper concludes in Sect. 5 with future research directions.

2 Related works

This section discusses the background, the existing methods in real-time video processing, and related areas. It is quite interesting and evident to note that very few research works have been carried out in recent years in this particular research domain. The works proposed in this research area are discussed along with its working methodology, advantages, and limitations. A camera with cohesive PAL module having VGA interface was proposed in [10]. Using the DSP + FPGA + SCM + ASIC architecture, it finishes the procurement of the video signal and display. This technique also focuses on major technologies of auto focusing and auto iris in FPGA. In the system, the image is enhanced by PW1226 and the frame is modified at 60 Hz. But the system suffers from few limitations and issues. The system is dependent much on the hardware. Also the introduction and usage of an exclusive single chip generate numerous complications which include increased cost and bulkiness. We use a smart configuration of FPGA to overcome these limitations in our proposed design and to achieve better performance.

Another RTVPS combining an FPGA chip with a DSP processor was proposed in [12]. The system used an FPGA chip as the major functional element for sampling the image and display. The DSP processor was utilized for critical image processing. The authors also tested the functionality of the system using an image edge detection algorithm. The system could incorporate the execution of the color model conversion and pixel-based conversion. But an FPGA-based system integrated with HDL can provide much more flexibility and robustness. This system would also have multiple advantages over the existing system. Authors in [13] proposed a system to process image using DSP and FPGA. Here the FPGA unit is used for sampling the image and display, and the DSP is the advanced processing module. The proposed system could acquire the image, process

the image using some operations like geometry transform, orthographic transform, operations based on pixels, image compression, and color space conversion. Although the system processed the images, the system is not flexible to incorporate real-time video processing.

Reference [14] proposed a color space conversion system based on FPGA. Here the system captured the video using a camera, and then obtained the YCbCr video data. This was then converted to RGB data by the system and at last displayed in the VGA format. This system was not flexible and suffered from few limitations in scalability. Few works have also focused on designing VGA display controllers based on FPGA and other extensions [15, 16], but they lack an efficient and flexible video conversion module.

Many works have also been done for the processing and display of images, but most of them are not scalable to incorporate the video processing module [17]. Many works have targeted on enhancing the security of transmitted images in the network [18–21]. An earlier basic version of the proposed system is available in [22]. Many novel and flexible modules have been added to this design.

In recent times, few major works using latest technologies like deep learning has been introduced to design efficient image and video processing systems [23–29]. Deep learning helps to improve the efficiency and performance of the image and video detection systems. Using various optimization techniques in deep learning, smart systems could detect and display the required image or video from a large data set within very short span of time. In our future work, we intend to use the deep learning technology to improve the efficiency of our real-time video processing system, and also introduce smart detection and tracking systems for fog networks [30].

3 System overview

This research article presents the results, analysis, and discussion from the implementation of an efficient method for converting video in standard PAL 576i format to standard video of VGA/SVGA format, in real time and implemented in FPGA with little utilization of resources, with VHDL used as the method of design entry for the entire system. The system has higher bandwidth to process video, enables rapid switch between multiple video inputs and buffers the real-time video output that is to be displayed. Character/text overlaying and several other applications also have been incorporated on the video.

One of the major tasks in any system is to decide the division of the modules between hardware and software. This is a very vital step deciding the performance of the implemented system. With additional functions to be added, a hardware–software co-design is highly essential in an embedded system [31].

4 System hardware–software co-design

This section presents and discusses the system hardware–software co-design. The major modules present in the video processing system are video capture, video control, function expansion, and video display modules. Figure 1 illustrates the proposed system with the various functional modules. The diagram also depicts the flow of control and data between the different modules in the proposed system. The video capture module uses a Charged Coupled Device (CCD) camera (PAL) to obtain the input video source. It then captures the video images, and then transmits back PAL format of the analog video signal. In the function expansion module, different design modules are developed and implemented in FPGA. It uses VHDL to achieve the corresponding system functionality. Functionalities such as color space conversion, video memory buffering, and character overlay are performed in this module. This module is flexible and we can add, modify or adapt any new features of the system in this module. SSRAM1 and SSRAM2 are used as video memories and act as a buffer between the video capture and VGA display module that is, two systems with different speed module. The video display module converts the digital image taken by FPGA into analog video information and outputs the information to display with VGA interface.

4.1 Video capture module

Figure 2 depicts the video capture module hardware. Here the CCD camera (PAL) is used as the input video source. It captures the video images, and then transmits back PAL format of the analog video signal for an A/D conversion. PAL is 625 lines, 50 fields per second, 2:1 interlaced monochrome standard used with television broadcasting standard. At a time, up to six video inputs can be given to the system between which video switching is also possible.

ADV7181 chip, a low-power high-speed multi-function digital video decoder chip is used in the system to complete the analog video to digital signal conversion. Along with decoding, video filtering should be done as part of image pre-processing. In the design of RTVPS proposed in [32], a chain of filters is incorporated along with other modules. The major limitations with real-time video filtering are in the increased delay and memory usage. These limitations directly affect the cost of the filter [12]. With single ADV7181 chip, it is possible to complete the video decoding along with all sorts of filtering and picture quality enhancement and it is also reliable.

ADV7181 has two stages; Analog Front End (AFE) and Standard Definition Processor (SDP). AFE digitizes

Fig. 1 System block overview

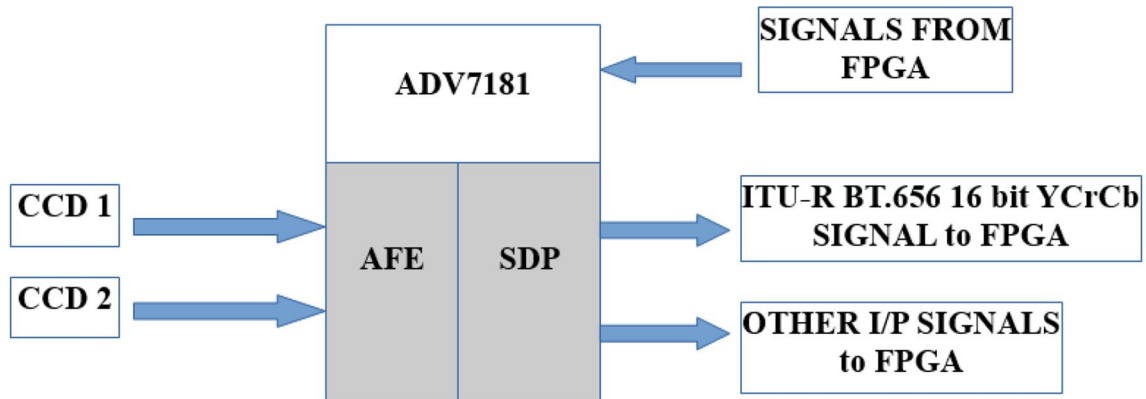
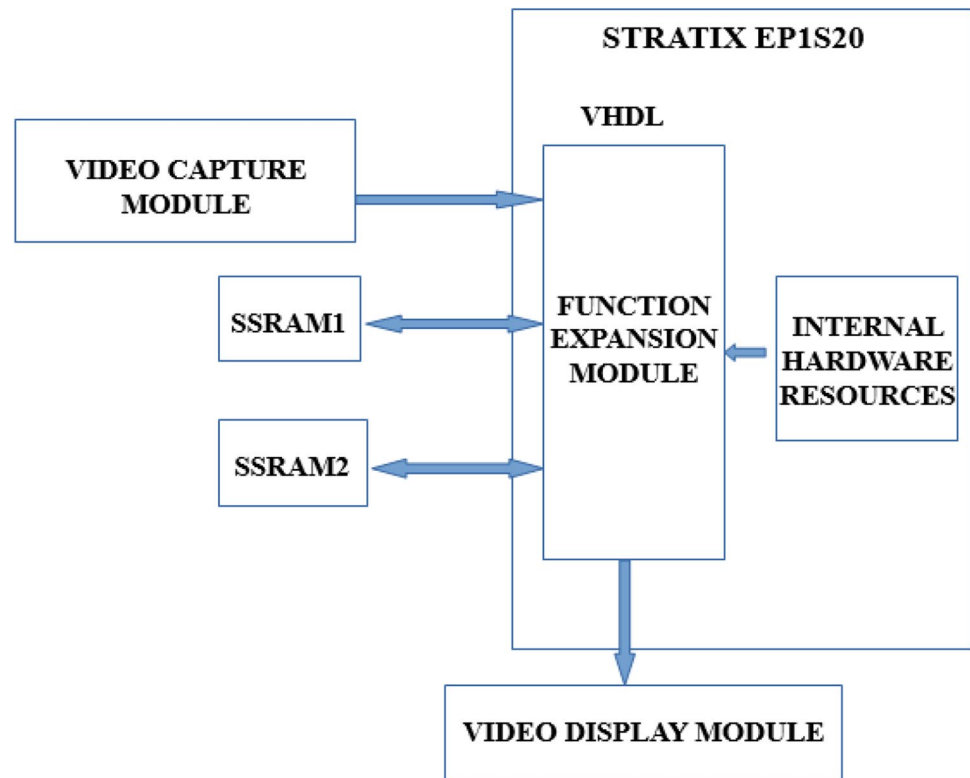


Fig. 2 Video capture module hardware diagram

the analog video signal along with fine clamping of the video signals before applying it to the SDP. The front end additionally has a 6-channel input mux that permits different video signals to be used with ADV7181B. It supports different analog input formats. All other functionalities are completed in SDP. It consists of comb filter that provides superior chrominance and luminance separation, a patented Adaptive Digital Line Length Tracking (ADLLT) algorithm to locate and decode poor quality video sources, Chroma Transient Improvement (CTI) processor and a Digital Noise Reduction (DNR) module.

4.2 Video control module and video memory

Real-time system requires creating a continuous and systematic reply to the environment which can be effectively achieved using FPGA and faster video memories. The received signal of YCbCr (4:2:2) format from the capture module is processed by FPGA. The FPGA used in the system is Stratix EP1S20. It is one of the most modern FPGA versions that operates at 1.5 V power supply and has 780-Pin FBGA package. It has 80 9×9 multipliers and has higher memory. This FPGA is ideal for a smart video processing

system. The system further uses an active crystal with 50 MHz working frequency. A good implementation of the video control module helps in the smooth functioning of the system and also reduces the access time.

This module is always associated with video memory. In this design, we require an external RAM to store the data. This is because large amount of data is generated with VGA display and the in-built RAM blocks of FPGA are unable to handle this. Multiple choices are available for memory usage in FPGA. A good design of the memory architecture would hugely develop accesses of memory in FPGA and also the utilization of FPGA in RTVPS [10].

Taking system memory size into account, VGA display system in [33] used one Static Random-Access Memory (SRAM). The major issue with this system was reading as well as writing cannot be executed together, because they were both planned for execution on the same chip. This reduced the speed of system due to the lack of parallelism. While typical design in [10] used two Synchronous Dynamic Random-Access Memory (SDRAM) in continuous full-page burst mode. Comparing SRAM and SDRAM as video memory, it is simpler to regulate the timing sequence of reading and writing with SRAM; SRAM offers multiple advantages compared to SDRAM. Additionally, SDRAM uses large amount of resources compared to SRAM [34].

In this design, two Synchronous SRAMs (SSRAMs) are used as video memories. System block overview diagram presented as Fig. 1 shows the SSRAM 1 and SSRAM 2 with the other modules. SSRAM acts as a buffer between the video capture and VGA display module that is, two systems with different speed.

4.3 Video display module

This module is designed to convert the digital image taken by FPGA into analog video information and output the information to display with VGA interface. VGA is a video transmission standard that follows progressive scanning. It also has numerous merits like increased resolution, quick speed in display, and rich color used in computer monitors. Figure 3 presents the hardware diagram of the video display module.

One of the major problems in using a general processor for video display controller is that, it does not have a separate module for processing graphics. It also would not be able to handle the huge data and can result in screen dithering and many errors while trying to display the image with different resolutions. Improvement and updating are also not feasible as it would lead to a lengthy development cycle leading to increased cost [35]. This would in turn affect the functionality of the modules and the efficiency of the system. Also using a specific chip designed only for this purpose would generate multiple problems in complexity of the design, increased cost, and bulkiness. Due to the above reasons, in our proposed system design FPGA acts as the VGA controller to realize the control of VGA display and each necessary module for VGA display is developed in the same process. This design is efficient and trims the size of circuit board considerably, thus also reducing the incurred cost of the entire project [36]. Different VGA display systems using Complex Programmable Logic Device (CPLD) or FPGA have already been developed [34, 36, 37]. Pixel data to the VGA display are in digital format.

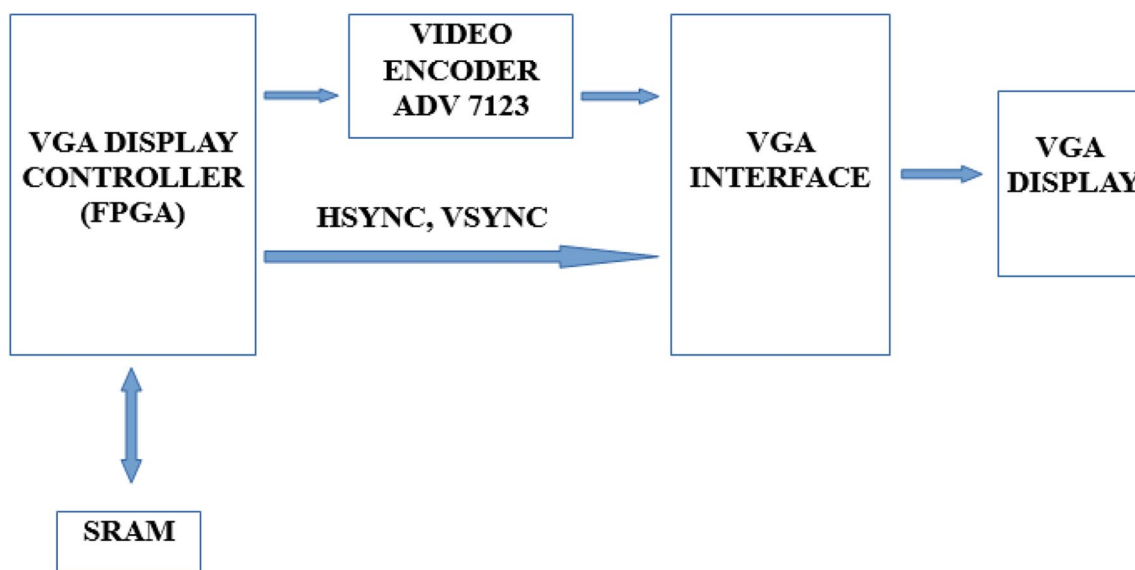


Fig. 3 Video display module hardware diagram

In Fig. 3, FPGA acts as the VGA display controller to realize the control of VGA display. A digital to analog conversion module between the FPGA and VGA interface is used. Here a unique video encoder chip ADV7123 is used to obtain the digital-to-analog conversion. The ADV7123 is a triple high speed, digital-to-analog converter. The analog output from DAC is given to the VGA monitor along with synchronization pulses generated by FPGA via VGA interface. Thus, the video is displayed in the monitor. An SRAM is used to store the characters and other texts for overlaying in the display.

4.4 Function expansion module

Different design modules that are developed and implemented in FPGA and that use VHDL to achieve system functionality are discussed in this module. This module is flexible and we can add, modify or adapt any new features of the system in this module.

4.4.1 Configuration of video decoder

The ADC, ADV7181B require a single 27 MHz clock which is provided by FPGA. ADV7181B should be configured accordingly for the module design. FPGA configures the decoder on the bus using I2C protocol. The interface uses two signals, serial data (SDA) and serial clock (SCLK) that are used to transfer data through the ADV7181B (slave) and FPGA (master). ADV7181B uses 249 subaddresses to access its internal registers, of which design needs only 16 register configurations. The remaining registers have default values after the system reset. The write sequence via serial data wire from FPGA used to configure ADV7181B is shown in Fig. 4.

In Fig. 4, the master begins the data transfer session with a start condition, which indicates that it will be followed by address/data stream. First byte indicates the address of the device and the second byte is the sub-address, followed by the data byte to be written to that corresponding internal register. Stop condition terminates the data transfer session. In between every byte, master waits for the acknowledgement. Thus, three bytes are required to configure one internal

register. Memory initialization file (MIF) is generated initially in which these three bytes (two address bytes + 1 data byte), each for 16 registers are recorded. This file is then saved to the FPGA on-chip ROM (1-Port). MIF file values are different for different video input channels due to the variation in internal register contents.

Design is also adapted to switch between different video inputs. Accordingly, different MIF files are generated and saved to on-chip ROMs. FPGA configures the video decoder by transferring the FPGA on-chip ROM data to the decoder input via I2C bus using the I2C protocol. Whenever video input channels are switched, corresponding on-chip ROM data are transferred immediately. The video decoder chip is configured to detect PAL standard composite video signal using the provided 27 MHz clock, and then converts it to a 16-bit video output data format. The pixel data output from the ADV7181B is sampled at its Line-Locked Clock (LLC) output at 13.5 MHz. The other output signals from ADV7181B are horizontal synchronization, vertical synchronization, and field signals.

4.4.2 Color space conversion

Phase- or frequency-modulating color subcarrier(s) with the color difference components, together produces the Y–C component form. The ‘Y’ component represents brightness and the ‘C’ component represents color. The color difference signals contain the difference between R (red) and the Y signal, and B (blue) and the Y signal. Also the BT.656YCbCr digital video uses 4:2:2 sampling. YCbCr is a color space used as a part of the color image pipeline in video and digital photography systems. Here Y is the luma component of the color. Luma component is the brightness of the color. In other words, it means the light intensity of the color. The human eye is more sensitive to this component. Cb and Cr are the blue component and red component related to the chroma component. Chroma samples coincide with alternate luma samples such as Cb, Y, Cr, Y, Cb, Y, Cr, etc. The sampling of luma is done at 13.5 MHz and sampling of Cb and Cr color difference components at 6.75 MHz. The interface is also known as 4:2:2, because luma is sampled at four times 3.375 MHz, and each of the Cb and Cr components

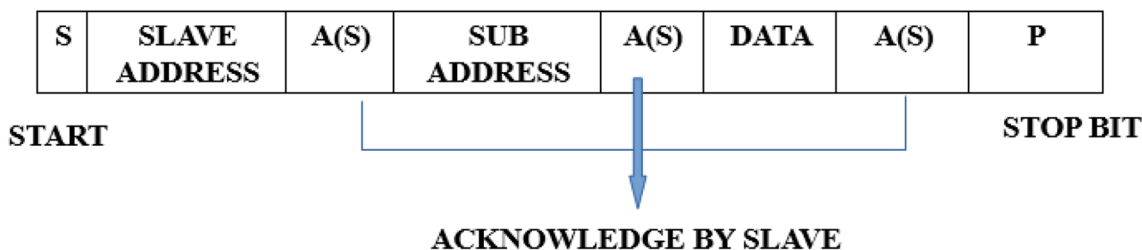


Fig. 4 Write sequence from FPGA to ADV7181B

at twice 3.375 MHz [8]. TV applications typically use color difference formats, whereas an RGB format is used in computer applications. Color space conversion is essential for transmission of information between devices that use distinct color space models. To transfer an image from television to computer, we need to transform the image from the YCbCr color space to the RGB color space [10, 38]. Therefore, YCbCr data format from the decoder is converted into RGB format to display it in the computer monitor or LCD panel using the following conversion equation [20]

$$\begin{aligned}
 R &= 1.164(Y - 16) + 1.596(Cr - 128) \\
 G &= 1.164(Y - 16) - 0.813(Cr - 128) - 0.391(Cb - 128) \\
 B &= 1.164(Y - 16) + 2.018(Cb - 128)
 \end{aligned}
 \tag{1}$$

4.4.3 Video memory buffering

The system uses two pieces of 1 M×24-bit SSRAM chip as the video memories to do ping pong buffering. In ping pong buffering, each memory performs a single operation for each time slot. The read and write operations in physically separate memory devices take place simultaneously. For real-time video, data buffering should be completed without any user perceivable delay. The major advantage of using ping pong over traditional memory devices is that, it allows the design of buffers operating twice as fast. This is one of the main benefits of RTVPS [39].

Figure 5 shows the memory allocation for read and write operations, writing to alternate memory locations while reading from continuous locations. The converted RGB data frame corresponding to the PAL in the interlaced format is written to memory1. For interlaced scanning, one frame is

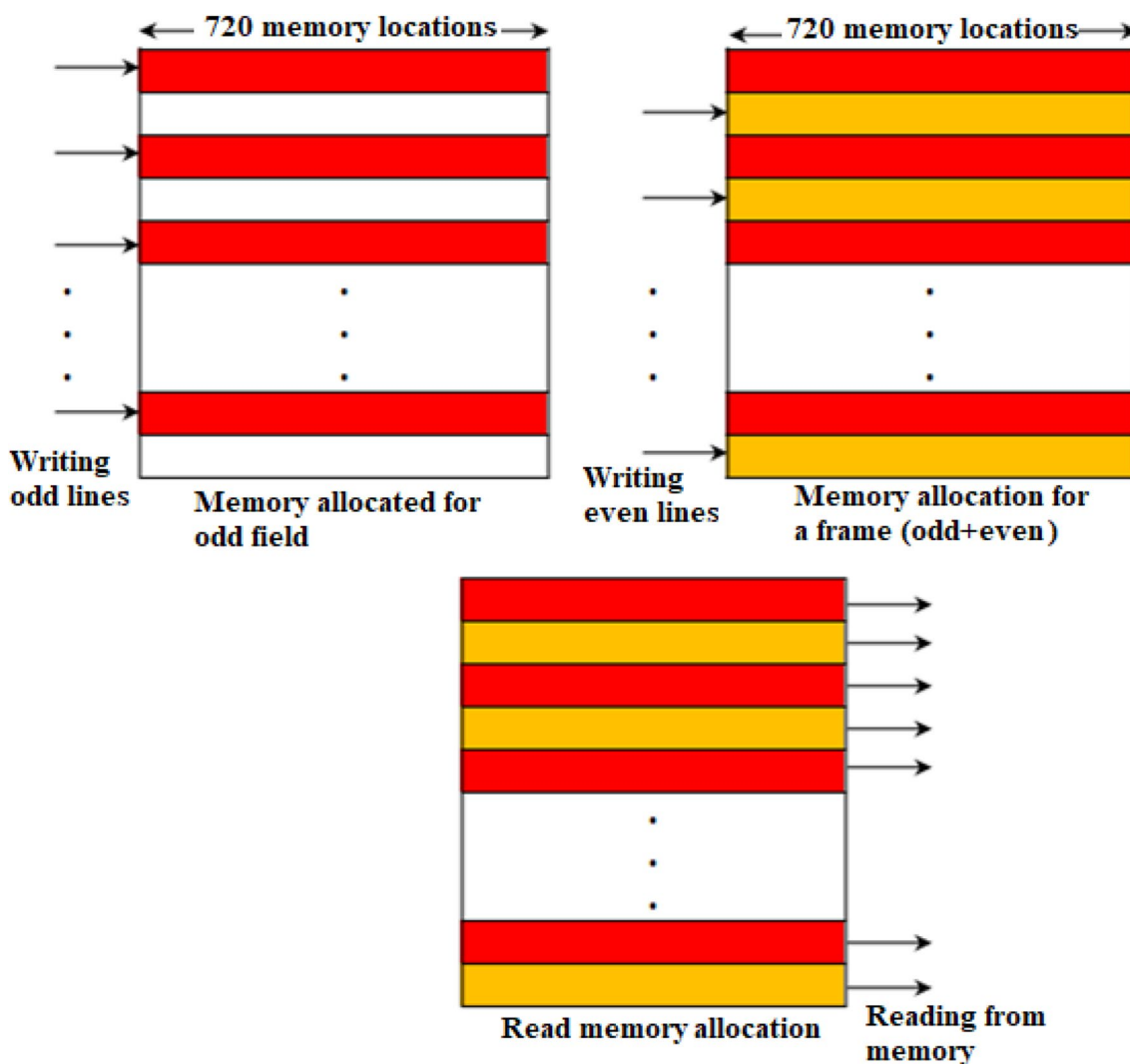


Fig. 5 Memory allocation

divided into two fields; odd field and even field. 720 active samples are present in each line in a frame. All the odd lines are allocated in the alternate 720 memory locations starting from base address, leaving space for the even field. For write operation, clock used is ADC output LLC with which each line of PAL video is sampled according to ITU-R_BT.656. While writing in memory1, reading is happening from memory2 in VGA progressive format. RGB data are read from continuous 640 memory locations by video display module starting from base address where the PAL data frames are stored. The reading clock given to memory is 25 MHz which is the pixel clock of VGA format 640×480 at 60 Hz.

The system memory structure is shown in Fig. 6. The read and write operations are switched between the memories for every generated frame. Neither of the memory is doing the same operation; while one frame is reading, next frame is writing simultaneously. It acts as interface between two different speed systems without any data loss. Thus, PAL to VGA conversion is achieved.

4.4.4 VGA display controller

Initially, from own video memory, FPGA processes out one line of data and then sends it to the inter-related pixels on the monitor and displays the following line. Once all the lines have been viewed, the next frame of image data will be displayed. Display refreshing rates of VGA monitor are the number of image frames displayed per second. Using high refreshing rate, illusion is created for human eye that makes the image continuous, rather than line by line [36, 40]. VGA and SVGA interface signal timings are shown in Table 1.

VGA interface consists of five signals, the line synchronous signal Hsync, field synchronous signal Vsync, and three RGB primary color signals. During line blanking or vertical blanking period, which is the retrace period, no active video is displayed. The three sections of blanking period are ‘front porch’, ‘line sync’ pulse, and ‘back porch’. FPGA acting as a VGA controller generates pixel clock signal, Hsync, Vsync, blanking signals by keeping right timing

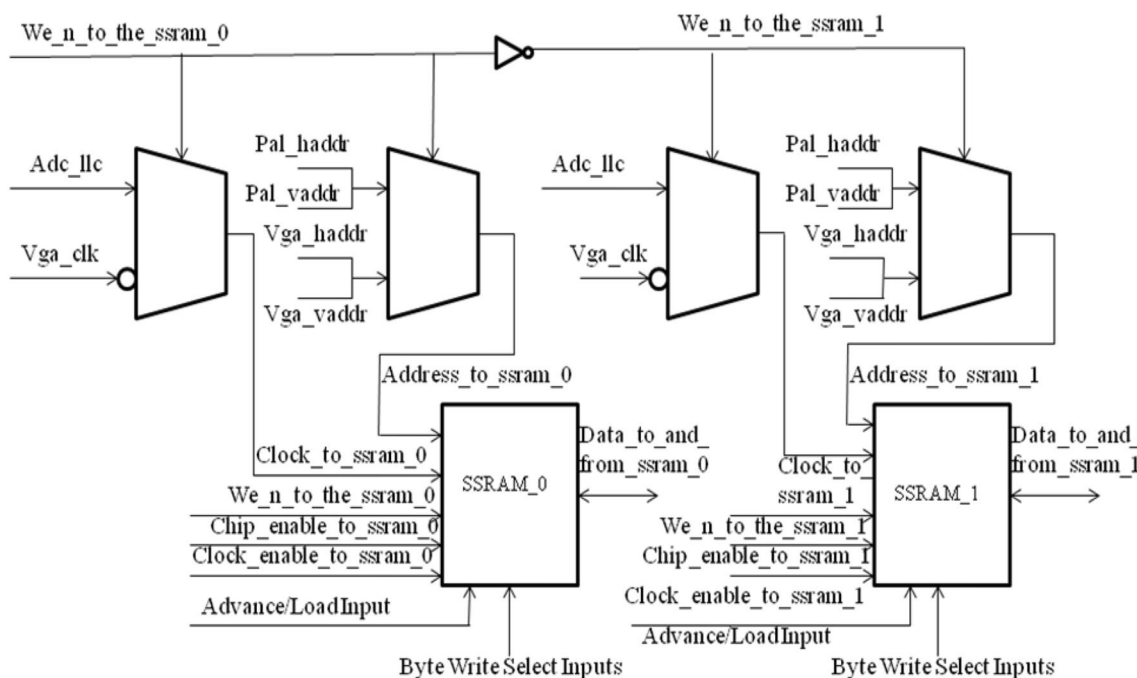


Fig. 6 Memory architecture of the system

Table 1 VGA timing

Format	Pixel clock (MHz)	Horizontal (in pixels)				Vertical (in lines)			
		Sync pulse	Back porch	Active video	Front porch	Sync pulse	Back porch	Active video	Front porch
640×480, 60 Hz	25.175	96	48	640	16	2	31	480	11
800×600, 60 Hz	40.000	128	88	800	40	4	23	600	1

relationships as VGA timing standard. Line and field counters are used to calculate effective area of image display and to produce the video memory read location from where RGB values corresponding to each pixel are obtained. Thus, obtained digital RGB signals are given to DAC, the analog red, green, blue primary signals integrated with FPGA generates Hsync and Vsync pulses that are sent to VGA interface. Thus, real-time video is obtained.

4.4.5 VGA/SVGA display

VGA is the video transmission standard defined in a range of resolution. Standard VGA resolution is 640×480 , Super VGA resolution 800×600 , extended VGA resolution is 1024×768 and so on. Our system is designed to display video either in 640×480 or 800×600 . In both cases, pixel clock signal, Hsync, Vsync, and blanking periods vary. To display in SVGA resolution, any of the two simple methods could be used, pixel duplication or nearest neighbor algorithm. For duplication, after each pixel interval, RGB value of previous pixel is duplicated. The value for an output pixel placed at (i, j) , the nearest neighbor method chooses the value of the nearest input pixel to $((i + 0.5) \text{ win}/\text{wout}, (j + 0.5) \text{ hin}/\text{hout})$. The computation processed by the scaler is equivalent to the integer calculation: $O(i, j) = F((2 \times \text{win} \times i + \text{win})/(2 \times \text{wout}), (2 \times \text{hin} \times j + \text{hin})/(2 \times \text{hout}))$. Here 'win' and 'hin' denote the width and height of the input image. The width and height of the output image are referred as wout and hout. The intensity value for a given point on the input image is found using function F , and function O produces an intensity value on the output image [38].

4.4.6 Character/text overlay

It is the technique in which character/text bypass the video or image displayed in the monitor. It is used in the system to display the title, some shapes, and digital clock and to scroll the text. Either FPGA on-chip RAM or off-chip RAM memory could be used for storing pixel information corresponding to the character to be overlaid. An SRAM chip is used for overlay character generation in the system. Data from SRAM are read to display in the corresponding pixel positions where character had to overlay. FPGA initiates the reading of data from SSRAM or SRAM progressively depending on whether video or text is to be displayed for that particular pixel streams and the final image is displayed in the monitor.

4.4.7 Skin color detection

Real-time skin color identification is very vital in a critical surveillance system. Skin color identification in YCbCr color space is chosen over RGB color space. RGB model

is not ideal since the primary color components are highly interconnected. Skin color region is efficiently extracted in YCbCr color space because Cb and Cr have some recognizable color span for skin area. Therefore, the algorithm works quite well. The skin color detection condition for YCbCr color space is given below

$$\begin{aligned} Y &> 60 \\ 85 &< Cb < 135 \\ 135 &< Cr < 180 \end{aligned} \quad (2)$$

The skin color identification is done on the video using YCbCr technique. The value is set to 1 or 0 depending on skin pixel or non-skin pixel. The system video output employs skin color identification.

4.4.8 Other applications

Using the VGA pixel information and clock overlay, a real clock with 1 s resolution to indicate time is designed. The clock also displays time from system power up or system reset. The time is shown on one row of the display. Luminance Y output obtained from the ADC is used to display monochrome display of the real-time video. Also, RGB values of each pixel are complimented to obtain negative video.

5 Implementation results

The VHDL codes are synthesized using Altera Quartus II design software and targeted for Altera Stratix 1S20 FPGA. The software package consists of HDL and schematic design entry, build and synthesis, power analysis and timing analysis, logic analyzer, and device schemes. After successful compilation, programming file is downloaded to FPGA via special connectors and the associated tools called as Universal Serial Bus (USB)-blaster which provides a USB interface to the host computer and Joint Test Action Group Universal Asynchronous Receiver Transmitter (JTAG UART) interface to the board.

5.1 Resource utilization

Tables 2 and 3 give the details of the components required for the design. Table 2 indicates the used, remaining, and usage percentage of the components. Our design requires only 11% of the total logic elements present in FPGA. On-chip memory utilization is also very low, less than 1%. Clock required for the various design modules are provided via Phase-Locked Loop (PLL). Even for further extension of the design, same clock can be used. It is quite evident from the results presented in Table 2 that the proposed system

Table 2 Resource utilization summary

Resource	Usage
Total logic elements	1995/18460 (11%)
Combinational with no register	1586
Register only	30
Combinational with a register	379
Logic element usage by number of LUT inputs	
4 input functions	1018
3 input functions	457
2 input functions	466
1 input functions	23
0 input functions	1
Logic elements by mode	
Normal mode	1460
Arithmetic mode	535
qfbk mode	51
Register cascade mode	0
Synchronous clear/load mode	148
Asynchronous clear/load mode	290
Total registers	409/21902 (2%)
Total labs	220/1846 (12%)
Logic elements in carry chains	587
User inserted logic elements	0
Virtual pins	0
I/O pins	205/587 (35%)
Clock pins	2/16 (13%)
Global signals	10
MS12s	3/194 (2%)
M4Ks	0/82 (0%)
M-RAMs	0/2 (0%)
Total memory bits	1402/1669248 (<1%)
Total RAM block bits	1728/1669248 (<1%)
DSP block 9-bit elements	0/80 (0%)
PLLs	2/6 (33%)
Global clocks	10/16 (63%)
Regional clocks	0/16 (0%)
Fast regional clocks	0/6 (0%)

utilizes very less resources. Total registers used in about 2% only which is quite efficient. Further total RAM block bits and memory bits used are less than 1%. So the proposed system has optimal utilization of the memory. There is 35% use of the I/O pins in the systems. This is necessary to obtain the required input to the system and also to display the converted output. This value is also good compared to the I/O utilization by already existing systems. The system also has smart and efficient usage of the clocks and is evident from the presented results.

Table 3 depicts the component usage of the design hierarchy from the top level entity. The number of components used by the distinct entity is denoted by the numbers in the

Table 3 Resource utilization hierarchically

Fitter resource utilization by entity		Fitter resource utilization hierarchically										
Compilation hierarchy node	LC register	Memory bits	M512s	M4Ks	M-RAMs	DSP element	Pins	LUT-only LCs	Register only LCs	LUT/register LCs	Carry chain LCs	Packed LCs
Video display	1995 (620)	409	1402	3	0	0	205	1586 (448)	30 (17)	379 (155)	587 (381)	51 (24)
Ycber to rgb:i5	303 (238)	177	0	0	0	0	0	126 (61)	10 (10)	167 (167)	170 (161)	24 (2)
adc_prog:i4	122 (122)	60	1024	2	0	0	0	62 (62)	3 (3)	57 (57)	36 (36)	3 (3)
rom1::inst1	0 (0)	0	512	1	0	0	0	0	0	0	0	0
rom1::inst2	0 (0)	0	512	1	0	0	0	0	0	0	0	0
dk_27:i2	0 (0)	0	0	0	0	0	0	0	0	0	0	0
graphics: i6	918 (918)	0	0	0	0	0	0	918 (918)	0	0	0	0
mode_sel:i7	32 (32)	0	0	0	0	0	0	32 (32)	0	0	0	0
pll:i1	0 (0)	0	0	0	0	0	0	0	0	0	0	0
room3:i9	0 (0)	0	378	1	0	0	0	0	0	0	0	0

parentheses. The total resources used by the specific entity and all of its subentities in the hierarchy are denoted by the numbers listed outside the parentheses. The presented results show that the proposed system has very low usage of resources compared to all existing systems. The color space conversion and video display modules utilize the available resources efficiently. It is also observed from the results that almost all the modules have free resources after the current usage. The video display module has almost or more than 50% free resources which is quite efficient. This helps the system to introduce more flexibility in the future. Additional real-time input may be obtained in future for processing using this system design. Only the graphics module has a higher usage. It is the module that does character/text overlay. From these results it's evident that the current design uses very less amount of resources and many more applications can be incorporated in future.

5.2 Thermal power analysis

The Altera Power Play Power analysis tool is used to estimate the device thermal power consumption. When design becomes complex and process technology continues to miniaturize, power consumption becomes a vital design consideration. Thermal power is dissipated as heat from FPGA. The outcome of the Power Play Power Analyzer is only an approximation of power dissipated. Summary of the analyzer by assuming default toggle rate of 12.5% for input signals is shown in Table 4.

The total thermal power dissipation is estimated as 510.93 mW. The core dynamic thermal power dissipation obtained in the system is 70.97 mW and core static thermal power dissipation obtained is 359.91 mW. The main source of the thermal power dissipated on chip is static power, which is independent of user clocks. It has

Table 4 Power Play Power Analyzer summary

Total thermal power dissipation	510.93 mW
Core dynamic thermal power dissipation	70.97 mW
Core static thermal power dissipation	359.91 mW
I/O thermal power dissipation	80.05 mW

Table 5 Thermal power dissipation by block type

Block type	Total thermal power by block type (mW)	Block and routing thermal dynamic power (mW)	Block thermal static power (mW)	Block average toggle rate (millions of transitions/s)
Logic cell	9.83	9.83	–	2.993
PLL	56.55	56.55	–	68.000
I/O	80.05	80.05	0.00	3.014
M512	4.59	4.59	–	6.250

heat dissipated by the routing and leakage power from all the FPGA modules. The I/O DC bias power is excluded in thermal power dissipation. Dynamic power is another main contributor for the power usage of the device resulting from signal variations. The I/O thermal power is due to the sum of I/O power taken by the VCCIO and VCCPD power supplies and power taken from VCCINT in the I/O submodules. It is quite evident from the results that the power usage by the proposed system is efficient and much lower compared to the existing standard systems.

Table 5 presents the calculated thermal dynamic and static power consumed by different block types. A vital statistic in estimating power consumption is the number of toggles per unit time. The toggles per unit of a signal are the average number of times that signal will change its value per unit of time. The commonly used unit is transitions per second, where a transition is a change from 1 to 0 or 0 to 1. From the results, it is quite evident that the proposed system has efficient usage of thermal power.

6 Conclusions

A novel RTVPS is designed using FPGA for efficient conversion of PAL to VGA, useful for rugged application with high bandwidth and little resource utilization. It can switch between multiple video input channels. Video output obtained is in real time and can display in VGA or SVGA format with character/text overlay. Entire functionalities and applications are developed using VHDL codes synthesized using Altera Quartus II design software and targeted for Altera Stratix 1S20 FPGA. The calculated FPGA resource requirements are recorded. The design is flexible and is possible to broaden or add many applications in this design. In our future work, we intend to use the deep learning technology to improve the efficiency of our real-time video processing system and also introduce smart detection and tracking systems for fog networks. Deep learning technologies could improve the efficiency of detection and display substantially.

References

1. Kehtarnavaz, N., Gamadia, M.: Real-time image and video processing: from research to reality. *Synth Lectur Image Video Multimed Process* **2**(1), 1–108 (2006)
2. Wenge, Z., Huiming, H.: FPGA-based video image processing system research. In: 2010 3rd International Conference on Computer Science and Information Technology, Chengdu, China, vol. 4, pp. 680–682. IEEE (2010)
3. Guo, X., Wei, X., Liu, Y.: An FPGA implementation of multi-channel video processing and 4 K real-time display system. In: 2017 10th International Congress on Image and Signal Processing, BioMedical Engineering and Informatics (CISP-BMEI), Shanghai, China, pp. 1–6. IEEE (2017)
4. Seyid, K., Richaud, A., Capoccia, R., Leblebici, Y.: FPGA-based hardware implementation of real-time optical flow calculation. *IEEE Trans. Circuits Syst. Video Technol.* **28**(1), 206–216 (2018)
5. Draper, B.A., Beveridge, J.R., Bohm, A.W., Ross, C., Chawathe, M.: Accelerated image processing on FPGAs. *IEEE Trans. Image Process.* **12**(12), 1543–1551 (2003)
6. Kechiche, L., Touil, L., Ouni, B.: Real-time image and video processing: method and architecture. In: 2016 2nd International Conference on Advanced Technologies for Signal and Image Processing (ATSIP), Monastir, Tunisia, pp. 194–199. IEEE (2016)
7. MacLean, W.J.: An evaluation of the suitability of FPGAs for embedded vision systems. In: 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)-Workshops, San Diego, CA, USA, p. 131. IEEE (2005)
8. Poynton, C.A.: *A Technical Introduction to Digital Video*. Wiley, New York (1996)
9. Lawal, N., O'Niels, M.: Embedded FPGA memory requirements for real-time video processing applications. In: 2005 NORCHIP, pp. 206–209. IEEE (2005)
10. Zheng, J., Gan, J., Chen, L.: The design of PAL integrative camera based on FPGA. In: 2010 International Conference on Multimedia Technology, Ningbo, China, pp. 1–4 (2010)
11. Altera Corporation: Stratix device family data sheet. *Stratix Device Handbook*, vol. 1 (2004)
12. Rao, M.G., Kumar, P.R., Prasad, A.M.: Implementation of real time image processing system with FPGA and DSP. In: 2016 International Conference on Microelectronics, Computing and Communications (MicroCom), Durgapur, India, pp. 1–4. IEEE (2016)
13. Jinghong, D., Yaling, D., Kun, L.: Development of image processing system based on DSP and FPGA. In: 2007 8th International Conference on Electronic Measurement and Instruments, Xi'an, China, pp. 2–791. IEEE (2007)
14. Zhang, X., Li, X., Yang, W., Li, R.: FPGA-based color space conversion system design and implementation. In: 2016 IEEE 7th Annual Ubiquitous Computing, Electronics and Mobile Communication Conference (UEMCON), New York, USA, pp. 1–4. IEEE (2016)
15. Dong, H., Guo, H.: Design of VGA display controller based on FPGA and VHDL. In: 2011 International Conference on Electric Information and Control Engineering, Wuhan, China, pp. 4125–4128. IEEE (2011)
16. Wang, G., Guan, Y., Zhang, Y.: Designing of VGA character string display module based on FPGA. In: 2009 International Symposium on Intelligent Ubiquitous Computing and Education, Chengdu, China, pp. 499–502. IEEE (2009)
17. Bouganssa, I., Sbihi, M., Zaim, M.: Implementation on a FPGA of edge detection algorithm in medical image and tumors characterization. In: 2016 5th International Conference on Multimedia Computing and Systems (ICMCS), Marrakash, Morocco, pp. 59–64. IEEE (2016)
18. Khosravi, M.R., Yazdi, M.: A lossless data hiding scheme for medical images using a hybrid solution based on IBRW error histogram computation and quartered interpolation with greedy weights. *Neural Comput. Appl.* **30**, 2017–2028 (2018)
19. Khosravi, M.R., Sharif-Yazdi, M., Moghimi, M.K., Keshavarz, A., Rostami, H., Mansouri, S.: MRF-based multispectral image fusion using an adaptive approach based on edge-guided interpolation. *J. Geogr. Inf. Syst.* **9**(02), 114 (2017)
20. Rajesh, S., Paul, V., Menon, V.G., Khosravi, M.R.: A secure and efficient lightweight symmetric encryption scheme for transfer of text files between embedded IoT devices. *Symmetry* **11**(2), 293 (2019)
21. Khosravi, M.R., Rostami, H., Samadi, S.: Enhancing the binary watermark-based data hiding scheme using an interpolation-based approach for optical remote sensing images. *Int. J. Agric. Environ. Inf. Syst. (IAEIS)* **9**(2), 53–71 (2018)
22. Alzubi, J., Jacob, S., Menon, V.G., Joseph, S., Vinoj, P.G.: A top-up design for pal to vga conversion in real time video processing system. In: 2018 International Symposium on Advanced Electrical and Communication Technologies (ISAECT), Rabat, Morocco, pp. 1–5. IEEE (2018)
23. Lan, X., Zhang, S., Yuen, P.C., Chellappa, R.: Learning common and feature-specific patterns: a novel multiple-sparse-representation-based tracker. *IEEE Trans. Image Process.* **27**(4), 2022–2037 (2018)
24. Lan, X., Ye, M., Shao, R., Zhong, B., Yuen, P.C., Zhou, H.: Learning modality-consistency feature templates: a robust RGB-infrared tracking system. In: *IEEE Transactions on Industrial Electronics*
25. Lan, X., Ma, A.J., Yuen, P.C., Chellappa, R.: Joint sparse representation and robust feature-level fusion for multi-cue visual tracking. *IEEE Trans. Image Process.* **24**(12), 5826–5841 (2015)
26. Lan, X., Ye, M., Zhang, S., Zhou, H., Yuen, P.C.: Modality-correlation-aware sparse representation for RGB-infrared object tracking. *Pattern Recognit. Lett.* (2018). <https://doi.org/10.1016/j.patrec.2018.10.002>
27. Jain, D.K., Zhang, Z., Huang, K.: Random walk-based feature learning for micro-expression recognition. *Pattern Recogn. Lett.* **115**, 92–100 (2018)
28. Jain, D.K., Zhang, Z., Huang, K.: Multi angle optimal pattern-based deep learning for automatic facial expression recognition. *Pattern Recogn. Lett.* (2017). <https://doi.org/10.1016/j.patrec.2017.06.025>
29. Ren, S., Jain, D.K., Guo, K., Xu, T., Chi, T.: Towards efficient medical lesion image super-resolution based on deep residual networks. *Signal Process. Image Commun.* (2019). <https://doi.org/10.1016/j.image.2019.03.008>
30. Menon, V.G., Prathap, J.: Vehicular fog computing: challenges applications and future directions. In: *Fog Computing: Breakthroughs in Research and Practice*, pp. 220–229. IGI Global, USA (2018)
31. Lawal, N.: *Memory synthesis for fpga implementation of real-time video processing systems*. Doctoral dissertation, Mid Sweden University (2009)
32. Chacko, B.T., Shelly, S.: Real-time video filtering and overlay character generation on FPGA. In: 2010 International Conference on Recent Trends in Information, Telecommunication and Computing, Kochi, India, pp. 184–189. IEEE (2010)
33. Fischer, V., Lukac, R., Martin, K.: Cost-effective video filtering solution for real-time vision systems. *EURASIP J. Appl. Signal Process.* **2005**, 2026–2042 (2005)
34. Shi, D., Ye, X.: Design of VGA display system based on CPLD and SRAM. In: 2013 Third International Conference on Intelligent System Design and Engineering Applications, Hiongkong, pp. 579–582. IEEE (2013)
35. Kalomiros, J.A., Lygouras, J.: A host co-processor FPGA-based architecture for fast image processing. In: 2007 4th IEEE

Workshop on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications, Dortmund, Germany, pp. 373–378. IEEE (2007)

36. Ziting, W., Haili, G., Yan, S.: Design of VGA image controller based on SOPC technology. In: 2009 International Conference on New Trends in Information and Service Science, Beijing, China, pp. 825–827. IEEE (2009)
37. Aihua, W., Dong, L., Zhen, W.: The design Of VGA data communication based on FPGA. In: 2011 IEEE International Symposium on IT in Medicine and Education, Guanzhou, vol. 1, pp. 649–651. IEEE (2011)
38. Altera Corporation: Broadcast video infrastructure implementation using FPGAs. White paper, ver. 1.2
39. Joo, Y.M., McKeown, N.: Doubling memory bandwidth for network buffers. In: Proceedings. IEEE INFOCOM'98, the Conference on Computer Communications. Seventeenth Annual Joint Conference of the IEEE Computer and Communications Societies. Gateway to the 21st Century (Cat. No. 98), San Francisco, CA, USA, vol. 2, pp. 808–815. IEEE (1998)
40. Altera Corporation: Video and image processing suite, User Guide (2009)

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Deepak Kumar Jain received the degrees of Bachelor of Engineering from Rajiv Gandhi Proudयोगी Vishwavidyalaya, India and Master of Technology from Jaypee University of Engineering and Technology, India in 2010 and 2012, respectively. He received his Ph.D. degree from University of Chinese Academy of Sciences, Institute of Automation, Beijing, China. He has presented several papers in peer reviewed conferences, as well as published numerous studies in science cited journals. His areas

of research are deep learning, machine learning, pattern recognition, and computer vision.



Sunil Jacob is currently the Director, Centre for Robotics, SCMS School of Engineering and Technology and also Professor in Department of Electronics and Communication Engineering. He has been the recipient of Young Gandhian Technological Innovation Appreciation Award 2018, and AICTE Chhatra Vishwakarma Award 2017 in Electronics. His project Muscles to Machine Interface for Paralyzed has been funded by IEEE EPICS

USA. Other funded projects include Bionic Haptic Arm, Rejuvenating the cells of human body, De-addiction coil for drug addicts, Smart Keyboard for Disabled Person, Low Cost 3D printer, Wearable device for detection and Prevention of Heart Failure, Bug-bot for Mosquito Attractor.



Jafar Alzubi received a B.Sc. (Hons) in Electrical Engineering, majoring Electronics and Communications from the University of Engineering and Technology, Lahore, Pakistan in 2001. In 2005 received M.Sc. (Hons) in Electrical and Computer Engineering from New York Institute of Technology, New York, USA. Between 2005 and 2008, he became a full time lecturer in the School of Engineering at Al-Balqa Applied University, Jordan. In 2008, He joined the Wireless Communica-

tions Research Laboratory at Swansea University, UK, completed his PhD in Advanced Telecommunications Engineering in June 2012. He is now an associate professor at Computer Engineering department, Al-Balqa Applied University, Jordan. His research interests include and focus on elliptic curves cryptography and cryptosystems, classifications and detection of web scams, using Algebraic–Geometric theory in channel coding for wireless networks. He is currently working jointly with Wake Forest University, NC, USA as visiting associate professor in Laboratory of Complex Brain Networks (LCBN). He has published more than 30 papers in highly respected journals. He is also managing and directing few projects funded by the European Union.



Varun Menon is currently an Associate Professor in Department of Computer Science and Engineering, SCMS School of Engineering and Technology and Head of International Partnerships at SCMS Group of Educational Institutions, India. He is also a distinguished speaker at Association of Computing Machinery (ACM). He is currently serving in the Review and Editorial Boards of 32 high impact factor journals including IEEE Transactions on Vehicular Technology, IEEE Transactions

on Communications, IEEE Communications Magazine. He has published 30 research papers in high-impact peer-reviewed international journals and conferences. He has also served in the Technical Program Committee and Organizing Committee of many prestigious international conferences organized in various countries like Thailand, USA, Hong Kong, Vietnam, etc. His research interests include Information Science, Scientometrics, Digital Library Management, Informatics of Scientific Databases, Researcher Parameter and Hijacked Journals Mobile Ad-Hoc Networks, Wireless Communication, Opportunistic Routing, Wireless Sensor Networks, Internet of Things, Fog Computing and Networking, Underwater Acoustic Sensor Networks, Educational Psychology, Cyber Psychology, Life Skills training, Training and Development.