

一种基于 ModelSim 的红外图像仿真方法与系统

李江辉

(上海热芯视觉科技有限公司, 上海 200331)

摘要: 在红外热成像的图像处理算法过程中, 传统的算法仿真需要借助 Matlab 等图形处理工具软件先进行仿真再转化成 FPGA 处理器中可以运行的算法代码, 而 Matlab 上的运行的语言、实现方式与 FPGA 硬件上运行的硬件描述语言和实现方式完全不同。导致转化过程复杂、转化精度损失、开发周期长等。本文提出了一种基于 ModelSim 仿真工具的红外图像仿真方法与系统, 与 Matlab 等图形处理工具一样, 编程写好代码, 导入图像仿真后, 立即输出图像显示, 并能查看中间处理过程中的数据变化情况。并且 ModelSim 仿真运行的仿真代码可以直接拿到 FPGA 编译工具中运行, 最终部署在硬件板卡上。在工程应用中, 转化过程简单, 开发效率大大提升。

关键词: FPGA; ModelSim; 红外热成像; 仿真方法; 图像处理算法; 硬件描述语言; 硬件加速

中图分类号: TN492 文献标识码: A 文章编号: 1001-8891(2024)07-0802-05

A Method and System for Infrared Image Simulation Based on ModelSim

LI Jianghui

(Shanghai Hot Core Vision Technology Co., LTD., Shanghai 200331, China)

Abstract: In the realm of image processing algorithms for infrared thermal imaging, traditional algorithm simulation often requires the use of graphics processing tools such as Matlab. These tools simulate the algorithm, which then needs to be converted into code that can run on FPGA processors. However, the language and implementation methods used in Matlab are entirely different from those used in FPGA hardware description languages. This results in complex conversion processes, loss of conversion accuracy, and long development cycles. This study proposes an infrared-image simulation method and system based on the ModelSim simulation tool. Similar to graphic processing tools such as Matlab, after programming and writing the code, one can import the image simulation, immediately output the image display, and view data changes during the intermediate processing stages. Furthermore, the simulation code run by ModelSim can be directly transferred to the FPGA compilation tool for deployment on the hardware board. In engineering applications, this simplifies the conversion process and significantly improves development efficiency.

Key words: FPGA, ModelSim, infrared thermal imaging, simulation method, image processing algorithm, hardware description language, hardware speedup

0 引言

在红外热成像设备的设计和应用中, 经常针对不同厂家或者不同型号的红外探测器定制非标的红外图像处理算法, 以及标定不同的算法参数。由于算法模型或者算法参数经常变化, 导致设计算法、仿真、在硬件上调试算法的时间很多, 开发周期很长, 开发效率低下等问题。而常规的红外图像处理算法, 包括数据采集、时域滤波、非均匀性校正、

空域滤波、锐化增强、坏点校正以及调光映射等。设计的初期, 需要从算法原理设计、算法模型的建立、数学过程的计算, 转化成可实现的编程语言, 然后实现和显示。

随着小型化、低功耗、低成本的红外热成像设备应用的普及和民用。FPGA 处理器非常适合于该种应用需求, FPGA 是高速并行的处理器, 适合于图像阵列的算法处理和加速, 并且功耗低, 可反复编程设计, 所以常应用于红外热成像设备中。

1 传统的红外图像仿真方法

基于 FPGA 的红外图像处理算法，仿真流程示意图如图 1 所示。

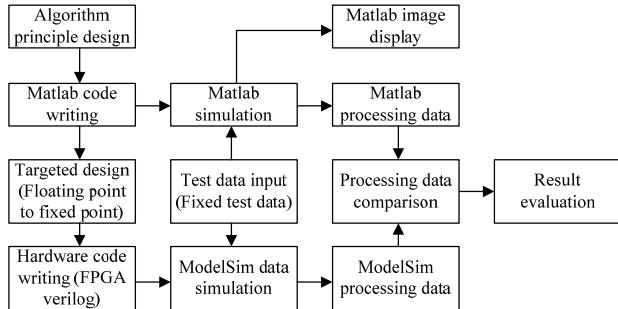


图 1 传统仿真方法示意图

Fig.1 Schematic diagram of traditional simulation methods

从图 1 中可知，利用 FPGA 处理器进行图像处理，需要两步的仿真结果。

第一步：根据算法需求，设计算法原理，在 Matlab 软件上编程实现算法过程，先用浮点数进行设计和计算，代码设计和编写完成后，导入测试图像，进行仿真和显示，在显示窗口中显示出来^[1-4]。对显示出来的图像效果进行评估，看看图像效果有没有达到算法原理期望的结果，如果没有，调整算法参数和实现过程，反复迭代调整直至图像效果达到期望的结果。达到期望结果后，进行定点化设计，将算法实现过程中的浮点数，全部转化成定点化数，比如 INT8、INT16、INT32 等，并且转化完成后，保证算法实现的结果仍然达到期望值，不损失精度，不丢失细节。然后再更换测试图像，测试数据设计成依次递增或者递减的特征数据，再进行仿真。

第二步：文献[5]、[6]提出定点化之后的算法原理和实现过程，编写 FPGA 的硬件描述语言，设计 FPGA 的逻辑，将第一步过程中的算法原理实现出来，再进行 ModelSim 仿真，仿真测试的输入激励是第一步中同样依次递增或者递减的特征数据，得到仿真结果，与第一步中每个关键节点（关键步骤）的结果进行比对，保证每一步的结果都与 Matlab 仿真的结果相同，才能确保算法效果的达到。

传统的红外图像仿真方法，分成上述两个步骤，先要进行 Matlab 仿真，显示评判效果达到之后，再启动定点化，将定点化之后的算法过程转化成 FPGA 中可以实现的过程，并用 FPGA 的硬件描述语言实现相关的逻辑，再进行 ModelSim 仿真，由于 ModelSim 不能像 Matlab 工具一样方便导入一张图像（图片数

据）进行仿真，仿真结果不能直观地可视化显示。所以在第一步、第二步过程中，输入激励的时候制作一些递增、递减的相同特征数据进行仿真。对每一步关键步骤和最后结果的仿真比对，保证数据正确，才能保证算法原理的实现和算法效果的达到。

2 基于 ModelSim 可视化的红外图像仿真方法

从上述传统的红外图像仿真过程看，既要实现 Matlab 代码编写、仿真、还要做定点化设计。等 FPGA 代码编写完成后，在 ModelSim 仿真时候还要进行每一步过程的计算结果比对，并且不能直接可视化的方式看到算法运行的结果。存在实现过程复杂、开发效率低下，开发周期长等问题。针对这些问题，本文提出了一种新的基于 ModelSim 可视化仿真的红外图像仿真方法。如图 2 所示为本文的实现框图。

从图 2 中可知，基于 ModelSim 可视化的红外图像仿真方法相比传统的红外图像仿真方法，仿真过程简单很多，开发效率也会提高很多。在算法原理设计完成后，直接编写 FPGA 的硬件描述语言实现算法计算过程，再进行 ModelSim 仿真，仿真的时候，不再用特定的递增数据或者递减的特征数据做输入激励，而是用之前原始采集到的红外图像，如果算法原理是针对 16 bit 的原始红外图像处理，可以是 Y16 数据，针对 8 bit 的调光之后红外图像处理可以是 Y8 数据，之前采集到的红外图像 Y16 或者 Y8 数据，存放在 RAW 文件中。而 RAW 文件中的图像数据是可以用 ImageJ 查看的，ImageJ 工具可以直接显示 16 位或者 8 位的 RAW 图像。

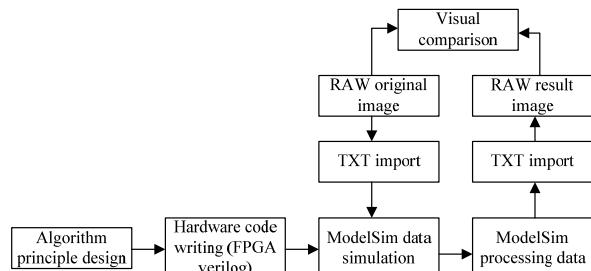


图 2 基于 modelsim 的红外图像仿真框图

Fig.2 Simulation block diagram of image based on modelsim

对于红外图像算法所用到的数据文件——RAW 文件，是不能直接导入 ModelSim 进行仿真的，需要经过 binary 工具将 raw 文件转换成 txt 文件，在 ModelSim 中通过系统调用\$readmemb/ readmemh 函数读入 TXT 文件的方式读入到内存中，然后进行图像算法处理，算法处理完成，调用 \$writememb/

\$writememh 函数将算法处理之后的图像写入 txt 文件中，通过 txt2bin 工具，将 txt 格式的数据转换成 bin 文件，再通过直接改后缀名，将.bin 改成.raw 文件。这样输入输出都是 raw 格式的数据文件，导入 ImageJ 工具中显示，以可视化的方式显示出了图像。

所以，这样红外图像算法处理的输入是图像数据，输出是处理之后的图像数据，可以直观地在 ImageJ 工具中显示出来对比，分析红外图像算法的处理过程是否正确。并且每一步的计算过程和关键步骤的结果也可以通过如图 3 内存查看的方式、图 4 内存读写过程仿真的方式查看数据在内存中存储、写入和读出是否异常。并通过如图 5 红外图像算法处理仿真过程来查看数据计算过程正确与否。

这样数据输入，写入内存，然后从内存中读出，再进行红外图像算法处理，每一步数据的处理都是可以查询和追溯的，所以既方便又快捷。如图 6 所示为本文进行红外图像滤波降噪算法处理过程中的输入源图和处理完成之后输出结果对比图 Image 显示，图 7 为本文中基于 ModelSim 仿真过程中输入输出文件结构示意图。

```
00000000 00000400 00000500 00000600 00000700 00000800 00000900 00000a00 00000b00 00000c00 00000d00
00000011 00001500 00001600 00001700 00001800 00001900 00001a00 00001b00 00001c00 00001d00 00001e00
00000022 00002300 00002400 00002500 00002600 00002700 00002800 00002900 00002a00 00002b00 00002c00
00000033 00003700 00003800 00003900 00003a00 00003b00 00003c00 00003d00 00003e00 00003f00 00004000
00000044 00004800 00004900 00004a00 00004b00 00004c00 00004d00 00004e00 00004f00 00005000 00005100
00000055 00005900 00005a00 00005b00 00005c00 00005d00 00005e00 00005f00 00006000 00006100 00006200
00000066 00006400 00006500 00006600 00006700 00006800 00006900 00006a00 00006b00 00006c00 00006d00
00000077 00007800 00007900 00007a00 00007b00 00007c00 00007d00 00007e00 00007f00 00008000 00008100
00000088 00008200 00008300 00008400 00008500 00008600 00008700 00008800 00008900 00008a00 00008b00
00000099 00009400 00009500 00009600 00009700 00009800 00009900 00009a00 00009b00 00009c00 00009d00
000000aa 0000a100 0000a200 0000a300 0000a400 0000a500 0000a600 0000a700 0000a800 0000a900 0000a000
000000ba 0000b100 0000b200 0000b300 0000b400 0000b500 0000b600 0000b700 0000b800 0000b900 0000b000
000000cc 0000c100 0000c200 0000c300 0000c400 0000c500 0000c600 0000c700 0000c800 0000c900 0000c000
000000dd 0000e100 0000e200 0000e300 0000e400 0000e500 0000e600 0000e700 0000e800 0000e900 0000e000
000000ee 0000f200 0000f300 0000f400 0000f500 0000f600 0000f700 0000f800 0000f900 0000fa00 0000f000
000000ff 00000301 00000401 00000501 00000601 00000701 00000801 00000901 00000a01 00000b01 00000c01
00000010 00001401 00001501 00001601 00001701 00001801 00001901 00001a01 00001b01 00001c01 00001d01
00000011 00001101 00001201 00001301 00001401 00001501 00001601 00001701 00001801 00001901 00001a01
00000012 00001101 00001201 00001301 00001401 00001501 00001601 00001701 00001801 00001901 00001a01
00000013 00003801 00003901 00003a01 00003b01 00003c01 00003d01 00003e01 00003f01 00003g01 00003h01
00000014 00004701 00004801 00004901 00004a01 00004b01 00004c01 00004d01 00004e01 00004f01 00004g01
00000015 00005801 00005901 00005a01 00005b01 00005c01 00005d01 00005e01 00005f01 00005g01 00005h01
00000016 00006901 00006a01 00006b01 00006c01 00006d01 00006e01 00006f01 00006g01 00006h01 00006i01
00000017 00007401 00007501 00007601 00007701 00007801 00007901 00007a01 00007b01 00007c01 00007d01
00000018 00008001 00008101 00008201 00008301 00008401 00008501 00008601 00008701 00008801 00008901
00000019 00009501 00009601 00009701 00009801 00009901 00009a01 00009b01 00009c01 00009d01 00009e01
0000001a 0000a101 0000a201 0000a301 0000a401 0000a501 0000a601 0000a701 0000a801 0000a901 0000a001
0000001b 0000b501 0000b601 0000b701 0000b801 0000b901 0000b001 0000b101 0000b201 0000b301 0000b401
0000001c 0000c101 0000c201 0000c301 0000c401 0000c501 0000c601 0000c701 0000c801 0000c901 0000c001
0000001d 0000e101 0000e201 0000e301 0000e401 0000e501 0000e601 0000e701 0000e801 0000e901 0000e001
```

图 3 红外图像算法处理内存数据

Fig.3 Infrared image algorithm processing memory data

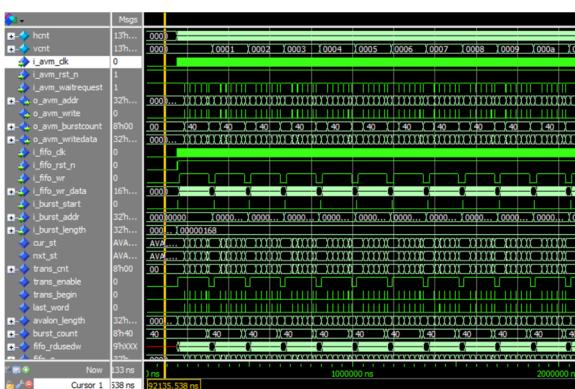


图 4 红外图像数据内存读写仿真波形数据

Fig.4 Infrared image data memory reading and writing simulation waveform data



图 5 红外图像算法处理仿真波形数据

Fig.5 Infrared image algorithm processing simulation waveform data



(a) The original input image (b) Output image

图 6 红外图像滤波降噪输入输出对比图

Fig.6 Infrared image filtering noise reduction input and output contrast diagram

名称	修改日期	类型	大小
result.raw	2022/5/26 20:15	RAW 文件	216 KB
result.txt	2022/5/26 20:15	文本文档	648 KB
source.raw	2021/10/18 11:12	RAW 文件	216 KB
source.txt	2022/5/26 20:08	文本文档	567 KB
Txt2Bin.exe	2010/12/13 14:43	应用程序	52 KB

图 7 基于 ModelSim 仿真过程文件结构示意图

Fig.7 Schematic diagram of file structure based on ModelSim

在反复调整算法过程，以及算法参数的情况下，能够很快得到想要的算法结果，比如图 6 中右边是降噪之后的效果图，降噪之后，图像噪声能消除了，但是图像变得模糊了，说明降噪算法参数设置不合适，通过调整滤波器的系数和阈值之后，降噪效果有明显的改善，如图 8 所示。

上述图 6 与图 8 中的红外图像滤波降噪算法为非局部平均滤波降噪算法，算法流程如图 9 所示。这种降噪算法是一种利用图像自相似性的空域滤波算法，相比传统高斯平滑、双边滤波等方法，保护更多的图像细节。图像各点之间可以同时进行运算，非常适合于在 FPGA 并行处理器上运算。根据相似性计算权重的时候，如果相似性阈值选择太大，算法滤波出来的效果如图 6 所示，过度平滑了，图像变模

糊了，而选择合适的相似性阈值，得到滤波出来的效果如图8所示，既能滤除噪声，又能保留图像细节。在该种红外图像仿真方法中能很快得到一组适合的算法参数。

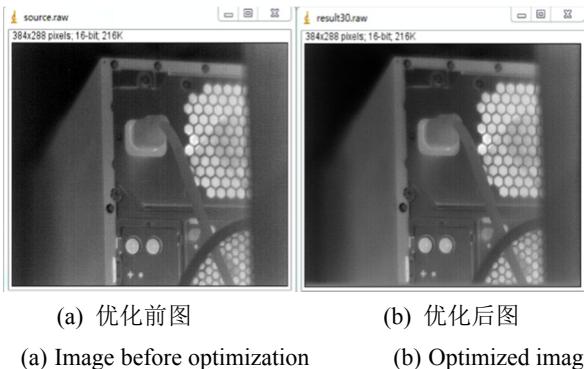


图8 红外图像滤波降噪后优化参数效果图

Fig.8 Optimization parameter effect after infrared image filtering and noise reduction

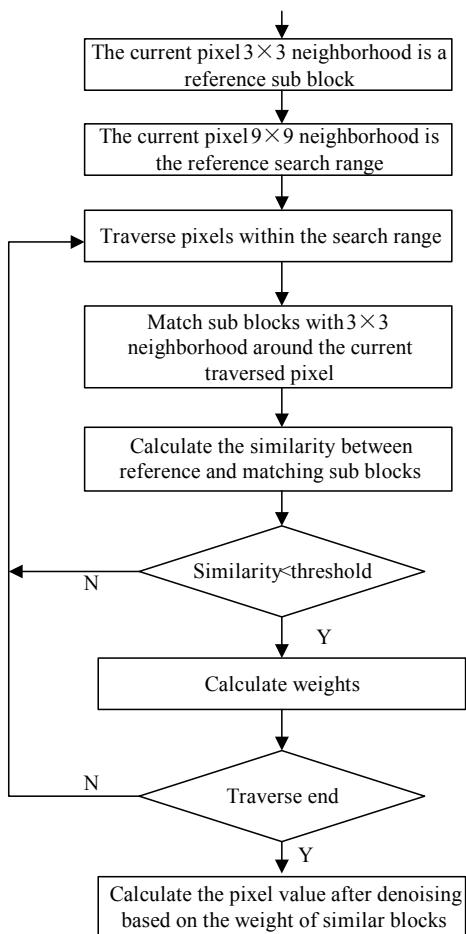


图9 非局部平均降噪算法流程图

Fig.9 Non local average noise reduction algorithm flowchart

基于该种 ModelSim 可视化的红外图像仿真方法，能够快速地修改算法参数和调整算法计算过程，运行之后得到图像算法的仿真结果，并且输

入、算法计算、输出等环节里的每一步数据都是可以通过内存或者波形查看的方式追溯数据处理过程，保证数据向着期望的方向计算和处理。

3 分析讨论

本文提出两种红外图像的仿真方法，传统的红外图像仿真方法，需要进行 Matlab 仿真，仿真结束后，再进行定点化设计，才能指导 FPGA 进行编程和实现，最后启动 ModelSim 的仿真，并且传统的 ModelSim 仿真并不能直接对图像数据的输入进行仿真并且可视化，而是制作相关的递增或者递减的特征数据作为输入，并且 Matlab 仿真同样以该特征数据作为输入，进行仿真，最后对 2 组仿真数据结果进行比对，比对结果一致代表该算法在 FPGA 中运行正常，能达到期望的算法效果。该过程复杂、繁琐、开发效率低下、开发周期长等问题不可忽视。所以本文在传统的红外图像仿真方法上进行了改进和优化，提出了一种新的基于 ModelSim 可视化的红外图像仿真方法与系统。该种方法在传统方法上省去了 Matlab 仿真过程和环节，同时也省去了定点化的设计过程。在算法原理设计完成后，启动 FPGA 硬件描述语言的编写和仿真，只是在仿真的时候，输入图像可以用之前采集好的红外图像或者用 PS 工具生成一张图片，保存为 raw 文件，在仿真输入的时候用 binary 工具将 raw 格式文件转换成 txt 文件，就可以启动仿真过程了，仿真结束后，生成的结果 TXT，再通过 txt2bin 工具，将 txt 格式的数据转换成 bin 文件，再通过直接改后缀名的方式，将 bin 改成 raw 文件。用 ImageJ 工具打开 source.raw 和 result.raw 文件就可以直观地对比显示在图像窗口上了。并且对于算法参数、计算过程的调整，也是很快得到相应的输出结果。该过程相对传统的 Matlab 仿真和 ModelSim 联合仿真要方便很多，尤其是在算法调整和参数的调整的情况下，又要重复一遍 Matlab 仿真，再去 ModelSim 仿真过程。但是该种方法对于红外图像算法原理的理解和 FPGA 逻辑实现图像算法的能力要求比较高。并且本文针对红外图像的仿真方法与传统的红外图像仿真方法，在适用范围上是一样的，对象都是红外图像的算法仿真。针对具体的算法模型，需要将算法分解成各计算步骤，然后针对各计算步骤进行数据仿真，所以对于其他文献 [7] 中不同的算法，比如弱小目标的背景的抑制，突出目标的增强算法等都是一样的仿真过程。

4 结论

对于红外图像处理算法在FPGA的实现过程中，本文阐述了传统的仿真方法，先进行Matlab仿真，仿真效果达到的前提下，再进行定点化设计，指导FPGA进行逻辑设计，完成算法原理的实现，最后启动ModelSim仿真，而且仿真结果的正确评判标准是与Matlab仿真结果比对一致，不一致的话，要优化定点化过程，再进行仿真，重复直至结果一致。而本文的仿真方法基于ModelSim的可视化仿真，省去了上述繁琐的开发过程，直接将图像源文件转化后导入ModelSim仿真，算法处理完成后，转成图像结果文件进行比对，用可视化的方式评判图像经过算法处理之后是否达到算法原理期望达到的算法效果。同时如果算法未到达期望效果，通过调整参数、算法实现过程，以及追溯输入图像、写入、读出内存过程，判断图像算法计算过程中数据的正确与否。所以该方法与传统的红外图像仿真方法相比，评估算法效果以可视化显示输入、输出对比图差异性来达到。评估算法计算过程的正确性，同样可以追溯数据处理过程的每一步的正确性来达到。所以该仿真方法的仿真效果是与传统的方法一致的。但是开发过程缩减了和效率上大大提升了，开发周期短。尤其是在反复调参和优化过程中，显得更为明显和方便便捷。

参考文献：

- [1] 刘凡, 肖树臣, 杨尚君, 等. 基于Creator和Vega的动态红外图像仿真研究[J]. 红外, 2011, 32(10): 21-26.
LIU Fan, XIAO Shuchen, YANG Shangjun, et al. Research on dynamic infrared image simulation based on creator and Vega[J]. *Infrared*, 2011, 32(10): 21-26.
- [2] 李波, 赵怀慈, 花海洋. 红外多光谱图像仿真方法研究[J]. 红外与激光工程, 2014, 43(6): 1972-1976.
LI Bo, ZHAO Huaici, HUA Haiyang. Infrared multispectral image simulation method[J]. *Infrared and Laser Engineering*, 2014, 43(6): 1972-1976.
- [3] 刘连伟, 杨淼淼, 邹前进, 等. 无人机红外辐射建模与图像仿真[J]. 红外与激光工程, 2017, 46(6): 628002-0628003(7).
LIU Lianwei, YANG Miaomiao, ZOU Qianjin, et al. UAV infrared radiation modeling and image simulation[J]. *Infrared and Laser Engineering*, 2017, 46(6): 628002-0628003(7).
- [4] 姚成喆, 郭伟兰, 陈钱, 等. 红外运动目标轨迹重构动态仿真平台[J]. 红外与激光工程, 2022, 51(2): 20210901.
YAO Chengzhe, GUO Weilan, CHEN Qian, et al. Dynamic simulation platform of infrared moving target trajectory reconstruction[J]. *Infrared and Laser Engineering*, 2022, 51(2): 20210901.
- [5] 唐麟, 刘琳, 苏君红. 红外图像噪声建模及仿真研究[J]. 红外技术, 2014, 36(7): 542-548.
TANG Lin, LIU Lin, SU Junhong. Modeling and simulation research of infrared image noise[J]. *Infrared Technology*, 2014, 36(7): 542-548.
- [6] 吴军辉, 王涛, 刘义和, 等. 红外图像的非均匀性及其仿真[J]. 红外技术, 2003, 25(5): 26-29.
WU Junhui, WANG Tao, LIU Yihe, et al. The nonuniformity of IR image and its simulation[J]. *Infrared Technology*, 2003, 25(5): 26-29.
- [7] 李飚, 徐智勇, 王琛, 等. 基于自适应梯度倒数滤波红外弱小目标场景背景抑制[J]. 光电工程, 2021, 48(8): 210122.
LI Biao, XU Zhiyong, WANG Chen, et al. Background suppression for infrared dim small target scene based on adaptive gradient reciprocal filtering[J]. *Opto-Electron Eng*, 2021, 48(8): 210122.