Development of New Image Processing Framework by the Collaboration of FUJIFILM and Fuji Xerox


Abstract

We developed a new image processing framework for different kinds of commercial products to reduce the development time and cost. Many of those products were independently designed for efficiency but without satisfying consumability and reusability, which have become increasingly important these days. To satisfy these requirements which have not been well considered, we have designed a new software framework based on the Pipes and Filters architectural pattern to orchestrate various combination of image processing modules. The framework provides a unified interface to easily develop modules to be orchestrated and flexibly customize the combination of modules in response to the various user requirements without significant performance degradation. It has been adopted by more than 20 products in the first two years and significantly reduced their development time and cost.

1. Introduction

To FUJIFILM (hereinafter referred to as FF), image processing technology is the source of product differentiation in photography, printing, medical care and various other fields. It is the fundamental technology and core technology to gain a competitive edge1). Achievements by those technologies are put into a group of our digital image processing software, called “Image Intelligence”. The Image Intelligence has added values to our products2).

Digital image processing technology is also the core technology for Fuji Xerox (hereinafter referred to as FX). FX has honed the technology for many years and created various added values particularly in the general office equipment market3).

FF and FX have decided that both could beef up the competitiveness and development efficiency by utilizing each other’s image processing technology. And, we have promoted a collaboration. However, the two companies’ technologies differ in specification and structure. To evaluate fairly and make the most of each other’s technology, we need to have a common basis for software to use as a mutually-agreed unified standard.

What is most required of the common software basis is versatility. The versatility here means applicability to the various business domains of both FF and FX. The existing technology of either company could not achieve sufficient versatility because of the following problems and trade-offs in pursuing versatility.

(1) Scalability of common software basis
It is difficult to expand common software basis to meet changes in operation environment or form of use. Such expansion requires a wide range of modifications.

(2) Ease of implementation of image processing functions
The structure is complicated and difficult to learn. It imposes a heavy workload on the developers in implementation.

(3) Efficiency and speed
If versatility is prioritized, an excess overhead and a decrease in speed may result, exceeding the tolerance level.

To solve the problems stated above and achieve versatility, we jointly developed the new common software basis, “FF/FX common image processing framework”4). A framework is a method of achieving reusability of software and a mechanism that provides the developers with a unified basis5).

In this report, Section 2 describes the software structures of FF and FX’s conventional technologies. Section 3 describes the structure of the FF/FX common image processing framework. Section 4 provides the evaluation of this framework on the problems stated above. Section 5 gives examples of application in products to show how much versatility is achieved.
2. Conventional Software Structure

Many of FF and FX’s existing image processing software programs (frameworks) employ the Pipes and Filters architectural pattern\(^6\). Similar patterns are widely used in other companies’ existing technologies. This report focuses on FF and FX’s existing technologies and aims at improving those technologies.

In many conventional technologies using this pattern, performance is put before versatility in order to solve the problem (3) mentioned in Section 1. The problems (1) and (2) have not been sufficiently taken into consideration. Especially the problem (2) is significant. As a result of pursuing efficiency and speed, the structure has become complicated and imposed a heavy workload on the developers.

Many of these conventional technologies are optimized for particular purposes. They conflict with the versatility we pursue.


3.1 Design Policy

As described in Section 2, the Pipes and Filters architectural pattern, which is used for many conventional technologies, is effective for a system that handles data as a stream. It is suitable for image processing, however, it also has a drawback. Because data are handled as a stream, if an error occurs during processing, it is difficult to resume processing (error reset). However, the drawback rarely seems to be a critical problem, since the architectural pattern has been used in many of FF and FX’s products and has shown successful results.

We have decided to use this Pipes and Filters architectural pattern for our new framework to achieve versatility by designing so as to solve the problems as described in Sections 1 and 2.

3.2 Overall Structure

Fig. 1 shows the structure of the new framework. The buffer module is the pipe and the image processing module is the filter of the Pipes and Filters architectural pattern. These are combined to build an image processing pipeline that does required processing. Next section provides detailed description of the main components of the framework shown in Fig. 1.

3.3 Component

3.3.1 Image Processing Module

If encapsulating an image processing task to be finally implemented as a module, we won’t be able to achieve versatility. Instead, we divide a required task into several single-function processing steps in accordance with the Pipes and Filters architectural pattern. We implement a series of image processing by combining those processing steps. Each of these processing steps is called the image processing module. Table 1 shows examples of image processing modules.

<table>
<thead>
<tr>
<th>Classification</th>
<th>Functions</th>
</tr>
</thead>
<tbody>
<tr>
<td>FF/FX common functions</td>
<td>Image format reading/writing (JPEG, TIFF, BMP, PNG, ...), geometric transformation, color conversion, gradation conversion, composition, filtering, compression, decompression, ...</td>
</tr>
<tr>
<td>FF function</td>
<td>Automatic image correction, facial skin smoothing, ...</td>
</tr>
</tbody>
</table>

Combining several of the image processing modules according to the purposes, image processing is implemented.

Each image processing module carries out processing for the set unit data size. The unit data size is freely set to suit the type of processing the module implements, for example, data for one pixel, one line, several lines or the whole image. We call this data size a block. Because each image processing module does processing block by block, optimization for each individual purposes is possible, for example, increase in speed and reduction in memory usage. That makes the internal structure simpler and imposes a less workload on the developers in implementation. The image processing modules, after processing block by block, accomplishes processing of the whole image.

3.3.2 Buffer Module

The buffer module controls data exchange between the adjacent processing modules. The buffer module supplies or stores only the block the image processing modules request. The blocks image processing modules process vary from module to module. However, image processing modules with different block settings can coexist in the same image processing pipeline, thanks to the buffer module. Encapsulating complicated data control in the buffer module eases the workload on the developers and makes the image processing module flexible and reusable.

3.3.3 Processing Management Unit

The processing management unit consists of the resource management unit, the error management unit and the workflow management unit. The resource management unit controls securing and releasing of memory space for each module. The error management unit controls information on
the errors caused during processing and action against those errors. The workflow management unit controls flow of the processing by the image processing modules. The processing management unit covers the whole image processing pipeline.

The management and control methods of each management unit are not fixed. They can be expanded to suit the user’s intension or the operation environment without having to make any changes in the program of the image processing module. The workflow management unit will be a good example. The following are examples of workflow management methods.

(1) Block-by-block workflow
Each image processing module passes over data for a block to next image processing module.

(2) Image-by-image workflow
Each image processing module processes data for the whole image and pass them to next image processing module.

(3) Parallel module workflow
The buffer module employs exclusion control. Each image processing module is assigned to a thread of execution and the image processing modules do processing in parallel.

4. Evaluation
We have evaluated this framework to see whether it overcomes the problems of the conventional technology described in Sections 1 and 2.

4.1 Scalability of Common Software Basis (Framework)
This framework has the processing management unit (resource, error and workflow management units) which covers the whole image processing pipeline, which the conventional technology does not have. These processing management unit makes it possible to expand the framework to meet changes in operation environment or form of use without any changes in the program of the image processing module.

4.2 Ease of Implementation of Image Processing Functions (Image Processing Module)
In the conventional technology, the block for the image processing module is fixed for the convenience of designing. That leads to a heavy workload on the developers in implementation. The image processing module of this framework is not subject to the restriction on block setting. The developers can select block setting freely. That helps keep the structure simpler. That also makes the implementation and optimization less demanding and reduces the workload on the developers in implementation of image processing modules.

4.3 Efficiency and Speed
As stated in Section 3, the image processing module and the buffer module of this framework are designed more versatile than the conventional modules. It is possible that the data control between these modules causes excess overhead and that a decrease in speed results, exceeding the tolerance level.

To check the overhead of data control between the image processing module and the buffer module, we have measured the processing time of several NOP (No OPeration) image processing modules linked together. Table 2 shows the two experiment environments. Fig. 2 shows the results of measurements using experiment images (a) and (b).

<table>
<thead>
<tr>
<th>Item</th>
<th>Configuration</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Environment A</td>
</tr>
<tr>
<td></td>
<td>Environment B</td>
</tr>
<tr>
<td>CPU</td>
<td>Intel Core2Quad Q6700 2.66 GHz</td>
</tr>
<tr>
<td>Memory</td>
<td>3.25 GB</td>
</tr>
<tr>
<td>OS</td>
<td>Windows XP Professional SP2</td>
</tr>
<tr>
<td>Compiler</td>
<td>Visual C++ 2005 SP1</td>
</tr>
</tbody>
</table>

![Figure 2](image.png)

In the design of the framework, we have assumed the tolerance of the excess overhead of data control to be below 10 msec for every image processing module in processing of the size of an image as shown in Fig. 2 (a) in the environment B in Table 2. We have calculated the tolerance from the capabilities of FF and FX’s conventional technologies and requirements by the product development departments.

As shown in Fig. 2, the excess overhead of data control is several msec. A decrease in the processing speed is within the tolerance.
5. Application in Products

5.1 Examples of Application

To date, this framework has been used in several tens of FF and FX’s products, for example, medical diagnostic equipment, color conversion software for printing, digital MFP and document management software. The framework is used for general purposes in a wide range of business fields, such as medical care, printing, photography and documentation. Not only applied in products, the framework is also being increasingly introduced to our core technology research departments. It is gaining ground as a common software basis for FUJIFILM Group.

5.2 Reduction in Development Man-Hours

Introducing this framework will increase reuse of image processing functions (image processing modules) and eliminate redundant development in FF and FX. That will reduce the development man-hours on image processing functions during product development.

We have calculated the development man-hours in product development by FF and FX in fiscal 2008. The number of development man-hours was reduced by approximately 120 man-months. We expect the development efficiency will be enhanced further.

6. Conclusion and Outlook

This paper has reported on the FF/FX common image processing framework, a common basis for image processing software used as FF and FX’s unified standard.

With FF and FX’s conventional technologies, if pursuing the versatility we require, we will have problems concerning scalability of the common software basis, ease of implementation of image processing functions, and efficiency and speed. To solve these problems, we have come up with a new common software basis, the FF/FX common image processing framework. This framework has been designed in accordance with the proven Pipes and Filers architectural pattern, which is used in the conventional technologies. The new framework has the following features.

* The processing management unit makes it possible to expand the framework to meet changes in operation environment or form of use.
* The processing data size (block) is freely set for each image processing module and reduces a workload on the developers in implementation.
* The excess overhead of data control between the image processing module and the buffer module does not exceed the tolerance level we have assumed. A decrease in speed is small.

The framework is used in FF and FX’s products in various business fields, verifying that the versatility has been achieved.

Although we did not go into details in this report, using the workflow management method (3) described in Section 3.3.3, it is possible to operate image processing modules in parallel. That means this framework is capable of working in the multi-core CPU environment, which is now in the mainstream. Further expansion is expected. We are planning to apply this framework in more products and services.

References


(In this paper, “Intel” and “Pentium4” are the registered trademarks of Intel Corp. “Windows” and “Visual C++” are the registered trademarks of Microsoft Corp. And “Image Intelligence” is a registered trademark of FUJIFILM Corporation.)