Acquisition Speed Comparison of Microscope Software Programs

OLIVER BIEHLMAIER, JOACHIM HEHL, AND GABOR CSUCS*
Light Microscopy Centre, Swiss Federal Institute of Technology (ETH), Zürich, Switzerland

KEY WORDS microscopy; software; acquisition time; performance

ABSTRACT Reliable software is a prerequisite for successful operation of a modern wide field fluorescence microscope. When used for live cell imaging, acquisition speed is of particular interest. This is both because biological processes can be highly-dynamic, and to avoid unnecessary photobleaching and phototoxicity of living samples. This article shows that besides the hardware (microscope) components themselves, the acquisition control software is an important influencing factor of speed performance. We tested and compared the speed performance of five different generic applications (Image-Pro Plus, MetaMorph, Micro-Manager, SlideBook, and Volocity) using typical experimental setups involving a single specific state-of-the-art fluorescence microscope configuration. The test measurements included multichannel experiments, *z*-stacking, burst acquisition, as well as combinations of these measurements with time-lapse acquisitions. The measured data provided values for guiding the testing and analysis of other microscope systems with similar configurations. Despite the identical hardware settings, significant and surprisingly large speed differences were evident among the various software applications. Additionally, no application was identifiable as the fastest in all tests. Our work pinpoints the importance of the control software in determining a system's "real" maximal imaging speed. The study could serve as basis for further tests, eventually influencing the system selection criteria for speed-sensitive applications. *Microsc. Res. Tech. 74:539–545, 2011.* © 2010 Wiley-Liss, Inc.

INTRODUCTION

Advanced light microscopy, especially fluorescence microscopy, is an essential technique in many biological disciplines. Over recent decades, an evolution has occurred of simple manual fluorescence microscopes into complex semiautomatic imaging stations. These stations give researchers almost unlimited potential for analyzing processes in living cells and tissue. A powerful computer is one prerequisite for success with light microscope systems. But another more-important prerequisite is reliable software, capable of controlling and integrating all microscope components and associated devices. The most important parameter of an optical system is arguably its sensitivity, but the speed (performance time) of the integrated system is also a critical factor, especially for experiments with living samples. This is not only because some biological processes are extremely fast (e.g., change of calcium status, blood flow, embryonic heart beat, etc.) (Vermot et al., 2008), but also to protect living samples from unnecessary photobleaching or phototoxicity effects. Although system speed is primarily a product of the underlying hardware components, the controlling software itself may also play an important role, as we show in the current paper.

Two alternatives exist when purchasing or configuring a state-of-the-art fluorescence wide-field imaging station. The first is to buy a complete system including the proprietary control software from a particular vendor. These may include systems with "real-time controllers" that drive hardware components directly, avoiding conventional operating systems such as Microsoft Windows. Examples include: CellR from Olympus; iMic from Till-Photonics; and DeltaVision from Applied Precision. A real time system allows more exact control and prioritization of hardware operations, and this certainly helps to optimize overall system speed. However, in almost all cases, user choice is restricted to a single software application and to a specific hardware configuration.

The second alternative is to separately purchase components (including generic software) and assemble the system by oneself. This latter method often occurs in academic environments, requiring systems that are flexible with hardware and software components. Aside from the fundamental choices in hardware, the selection of a particular generic software application to run the system is also critical. Many possibilities exist, although particular hardware configurations (like in our case) may limit the available choice.

There have been several reports evaluating the microscope system performance: wide-field fluorescence (Murray, 1998); confocal microscopes (Zucker and Price, 2001); comparison of wide-field systems and laser-scanning confocal systems (Swedlow et al., 2002; Murray et al., 2007); and comparison of single point-

Oliver Biehlmaier and Joachim Hehl contributed equally to this work.

*Correspondence to: Gabor Csucs, Light Microscopy Centre, Swiss Federal Institute of Technology (ETH), Schafmattstrasse 18, 8093 Zürich, Switzerland. E-mail: csucs@lmc.biol.ethz.ch

Received 27 May 2010; accepted in revised form 27 August 2010

DOI 10.1002/jemt.20944

Published online 17 November 2010 in Wiley Online Library (wileyonlinelibrary.com).

and spinning disc confocal systems (Wang et al., 2005). These reports focused mainly on evaluating efficiency in terms of signal-to-noise ratio and image quality. In doing so, they ignored performance-time related issues, and did not consider the influence of the controlling software on system speed.

In our article, we tested and compared the performance-time of five different generic applications for microscope control. Characteristic experimental setups were employed, all involving a single specific state-of-the-art fluorescence microscope configuration. While the selection of software applications to compare is somewhat arbitrary, we nevertheless followed two criteria: first, that the software should officially fully support our configuration; and second, it should be considered as well-established general-purpose control software in Europe and North-America. The results of this study could potentially serve as a baseline for comparing the performance time of wide-field imaging systems, and raise important questions regarding the future direction of control software development. The study could also potentially serve as a decision-making basis for selecting control software for speed sensitive applications.

MATERIALS AND METHODS

System Setup

We used a Zeiss Axiovert 200M stand equipped with the standard motorized *Z*-drive (focusing speed: 125 ms/1 μm), an excitation and emission filter-wheel (changing time for adjacent filter-wheel positions according to the manufacturer: 55 ms; for the loaded wheel in our test system: 70 ms) from Prior controlled by a Prior ProScanII controller, and a Uniblitz fluorescence shutter from Vincent Associates controlled by a Vincent Associates VMM-D1 shutter driver (Fig. 1). The particular shutter was employed because it enables faster image acquisition—with a shutter closing time of 9 ms compared with 50 ms with the Zeiss-FL-shutter included in the Axiovert stand. All three components were connected to the acquisition computer by serial COM-ports (RS232 standard). A Hamamatsu C8484 camera directed by a Hamamatsu A3472-06 controller performed image capturing (readout time: 112 ms = 8.9 f/s). All components were connected to a Fujitsu Siemens CelsiusM running on Microsoft Windows XP Professional (Version 5.1.2600 Service Pack 3 Build 2600) equipped with a 1866Mhz Intel processor, 4GB RAM, and a NVIDIA Quadro FX 1500 with 256MB of memory.

Tested Software

The following software applications were tested and compared with respect to their performance time: MetaMorph 7.6.3 (Molecular Devices, USA), Slide Book 4.2.0.12 (Intelligent Imaging Innovations, USA) Micro-Manager 1.3 (developed by Arthur Edelstein, Nico Stuurman and Nenad Amodaj, University of California, San Francisco), Volocity 5.2 (PerkinElmer, USA), and Image-Pro Plus AMS 7.0 (Media Cybernetics, USA). Each software application was installed according to the manufacturer's instructions. Among the software applications, only Micro-Manager is maintained as an open-source (free) product. The other four applications are commercial.

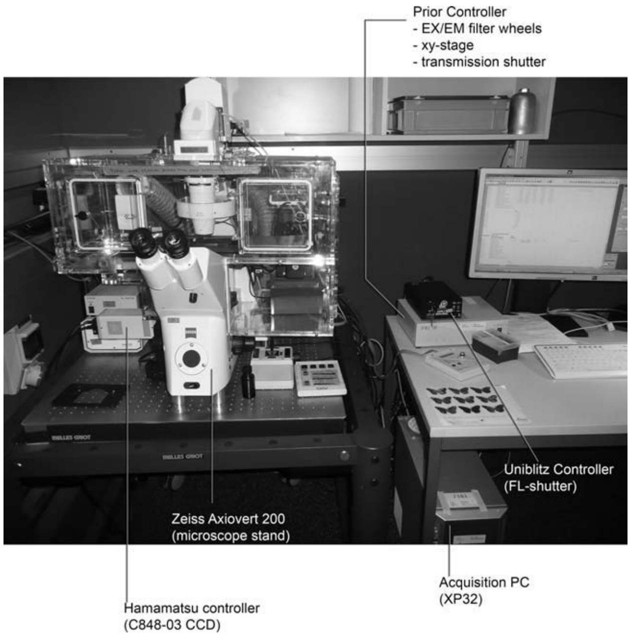

Fig. 1. Microscope system setup as employed for the performance tests. Image acquisition via a Hamamatsu C8484-03 camera, connected to the side port of a Zeiss Axiovert 200m stand. A Prior controller was used to direct Prior excitation and emission filter wheels, along with a Prior *xy*-stage, and a Prior transmission shutter. A Vincent shutter controller was used to direct the fast fluorescence shutter.

Testing Procedure

To test and compare the performance-time of these five different control applications, nine typical experimental setups were designed (see Table 1). The performance times of the experiments were calculated via the time stamp functionality of each application. If this was not possible directly, we extracted equivalent information from log files. Every measurement was repeated five times in order to additionally consider the reproducibility and stability of system performance.

RESULTS

Streaming/Burst Acquisition

High-speed image acquisition is crucial for live cell experiments in which the frame rate of the imaging system determines the success of the imaging experiment. A common solution for the fastest possible frame rate is to use stream- or burst-acquisition. Usually, this is only possible in one channel mode without a need to change filters. In addition, shutters usually remain open during the acquisition procedure. The only delimiters of the potential acquisition time are thus the readout time of the camera and the data handling time of the software application (e.g., RAM buffering). We tested all acquisition programs in this fast imaging mode—imaging 20 time points in a single channel at maximum speed with an open shutter (Experiment 1, Fig. 2a) or a remotely closing shutter (Experiment 2, Fig. 2b) to simulate maximum sample protection.

In Experiment 1, the fastest software application was MetaMorph with an acquisition time of 2.1s followed by Micromanager with 2.4 s. Velocity required 2.5 s, SlideBook required 3.2 s, and Image Pro 7 was the slowest program with 3.8 s (Table 2). Experiment 2

TABLE 1. Experimental setup

	Exp 1	Exp 2	Exp 3	Exp 4	Exp 5	Exp 6	Exp 7	Exp 8	Exp 9
Experiment description	1 ch burst/stream acquisition	1 ch time lapse	2 ch fast time lapse	2 ch slow time lapse	1 ch fast *z*-stack	1 ch slow *z*-stack	2 ch fast *z*-stack	2 ch z-stack z first	2 ch z-stack ch first
Channels	1	1	2	2	1	1	2	2	2
Shutter	open	remote	open	remote	open	remote	open	remote	remote
Time points	20	20	20	20	None	None	None	None	None
z-stack	None	None	None	None	10 slices	10 slices	10 slices	10 slices	10 slices

Exposure time was always 50 ms/channel. Emission and excitation filters were on adjacent filter wheel positions. If *z*-stacks were taken, step size was 1 μm.

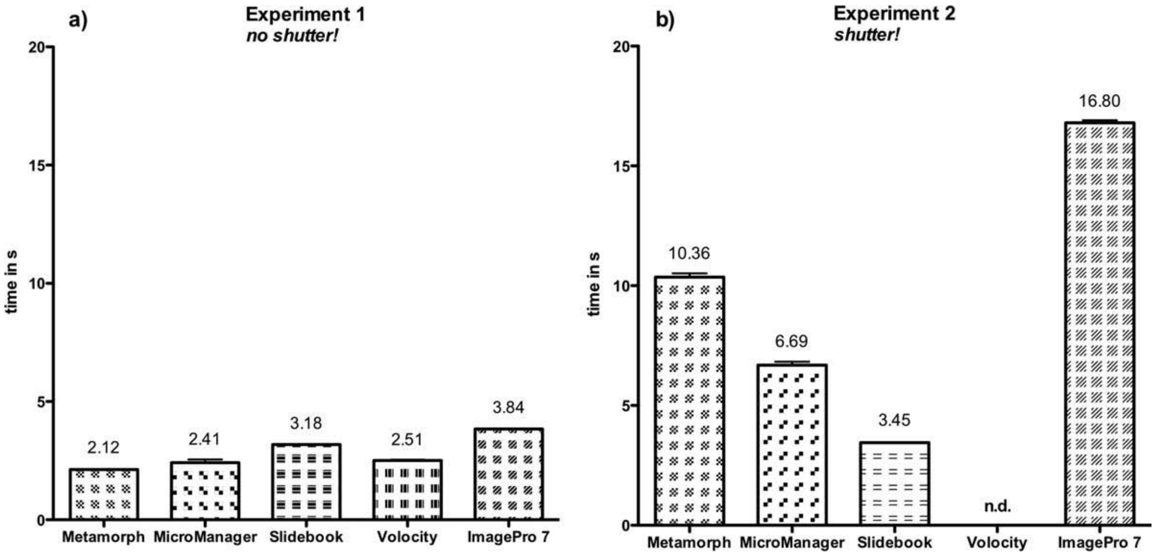

Fig. 2. Performance time comparison for a one channel time lapse with 20 time points at maximum speed. (**a**) Experiment 1: burst/stream acquisition performance of the five programs tested without shutter closing between frame acquisition. (**b**) Experiment 2: acquisition performance with standard settings including shutter closing. n.d., nondetermined.

(identical but with remotely closing shutter) resulted in 3.4 s with SlideBook, 6.7 s with MicroManager, 10.4 s with MetaMorph, and 16.8 s with ImagePro7 (Table 2). As Volocity was not able to remotely control the shutter during a stream- or burst-acquisition, no data were available for that program in Experiment 2.

Multichannel Time-Lapse Acquisition

Another typical application in live cell imaging is multichannel time-lapse acquisition. Accordingly, we tested in Experiment 3 (Fig. 3a) and Experiment 4 (Fig. 3b) how fast each application can change filter-wheels in a two-channel time-lapse acquisition experiment. We employed emission and excitation filters on adjacent filter wheel positions. Acquisition time was again measured for a 20 time point time-lapse experiment with open shutter (Experiment 3, Fig. 3a) or remotely-closing shutter (Experiment 4, Fig. 3b).

In Experiment 3, ImagePro completed the task in 14.3 s as the fastest application. MicroManager followed with 15.2 s, SlideBook with 17.4 s, Volocity with 20.7 s and finally MetaMorph with 27.7 s (Table 2). Remotely closing the shutter (Experiment 4) provided the following results: MicroManager 15.9 s, SlideBook 17.4 s, Volocity 24.5 s, MetaMorph 27.8 s and ImagePro7 47 s (Table 2).

Z-Stack Acquisition

Three-dimensional reconstruction and colocalization analysis are examples of applications of live cell imaging that require *z*-stack acquisition. In Experiment 5 (Fig. 4a) and Experiment 6 (Fig. 4b), we tested performance time in a single channel experiment - acquiring 10 slices with a step size of 1 μm. As before, we checked the respective performance times with an open shutter (Experiment 5) and a remotely closing shutter (Experiment 6).

SlideBook solved the task with an open shutter (Experiment 5) within 2.6 s, ImagePro7 needed also 2.6 s, Volocity 3.3 s, MicroManager 5.6 s, and MetaMorph 5.8 s (Table 2). With remote closing of the shutter (Experiment 6), SlideBook finished after 2.8 s, Volocity after 4 s, MicroManager after 5.6 s, MetaMorph after 5.7 s, and ImagePro7 after 6.3 s (Table 2).

Multichannel *z*-Stack Acquisition

Our final set of tests combined multichannel and *z*-stack acquisition. We consider this experimental setup representative of the majority of multidimensional live cell imaging experiments in a standard life science laboratory.

In Experiment 7 (Fig. 5a) we opened the shutter during the whole acquisition. Initially, we acquired the

TABLE 2. Measurement results

	Calc. speed	MetaMorph	Micro Manager	SlideBook	Volocity	ImagePro 7
Exp 1	2.50	2.12 ± 0.00	2.41 ± 0.14	3.18 ± 0.00	2.51 ± 0.03	3.84 ± 0.02
Exp 2	2.86	10.36 ± 0.16	6.69 ± 0.14	3.45 ± 0.00	n.a.	16.80 ± 0.11
Exp 3	3.83	27.66 ± 0.05	15.21 ± 0.85	16.81 ± 0.03	20.71 ± 0.15	14.27 ± 0.42
Exp 4	4.19	27.85 ± 0.19	15.89 ± 0.21	17.36 ± 0.00	24.47 ± 0.12	46.97 ± 0.05
Exp 5	2.95	5.79 ± 0.09	5.62 ± 0.03	2.64 ± 0.00	3.32 ± 0.57	2.64 ± 0.01
Exp 6	3.13	5.74 ± 0.08	5.63 ± 0.03	2.76 ± 0.00	3.99 ± 0.07	6.30 ± 0.11
Exp 7	5.97	10.65 ± 0.10	11.81 ± 0.30	9.38 ± 0.02	6.40 ± 0.16	4.95 ± 0.03
Exp 8	6.33	10.65 ± 0.06	11.87 ± 0.11	8.30 ± 0.23	8.48 ± 0.06	19.15 ± 0.06
Exp 9	6.89	13.66 ± 0.03	9.62 ± 0.03	9.36 ± 0.02	14.44 ± 0.08	20.55 ± 0.09

Calculated performance times (sum of times from system components) together with mean values ($n = 5$) and standard deviations of the respective measurements. One way ANOVA P-value < 0.0001 for all experiments. n.a., not available.

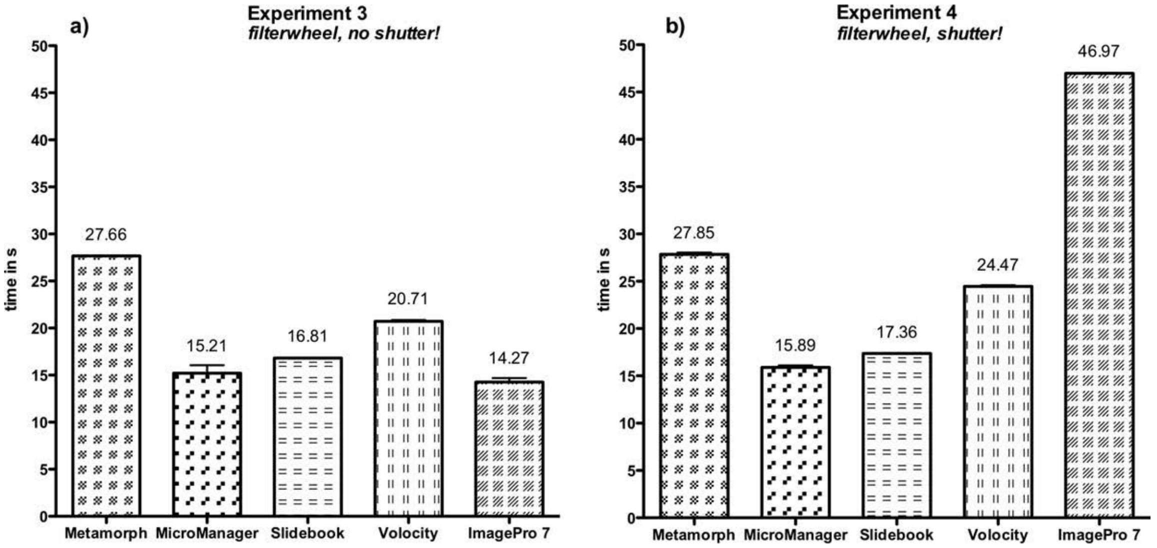

Fig. 3. Performance time comparison for a two channel time lapse with 20 time points. (**a**) Experiment 3: fast acquisition performance of the five programs tested without shutter closing between frame acquisition. (**b**) Experiment 4: acquisition performance with standard settings including shutter closing.

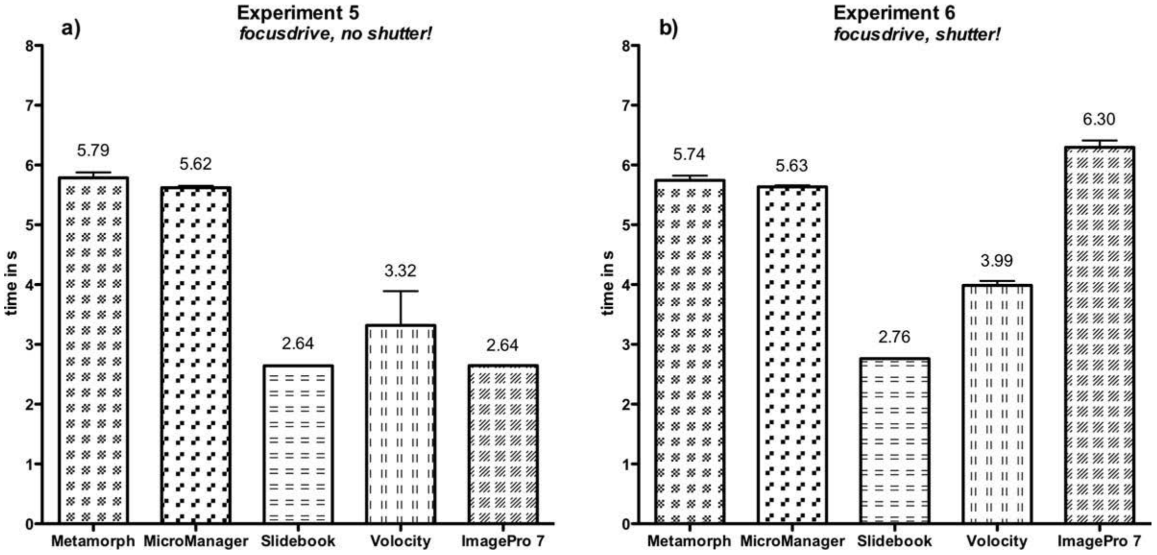

Fig. 4. Performance time comparison for a z-stack experiment with 10 slices. (**a**) Experiment 5: fast acquisition performance of the five programs tested without shutter closing between frame acquisition. (**b**) Experiment 6: acquisition performance with standard settings including shutter closing.

z-stack for Channel 1, subsequently we captured the *z*-stack for Channel 2 (“between stacks” configuration). In Experiment 8 (Fig. 5b) and Experiment 9 (Fig. 5c) we remotely closed the shutter. These latter experiments had different capture sequences for the channels and *z*-stack. In Experiment 8, as in Experiment 7, we used the “between stacks” mode, and in Experiment 9 we initially captured the two respective channels for each slice and then switched to the next frame (“between frames” configuration).

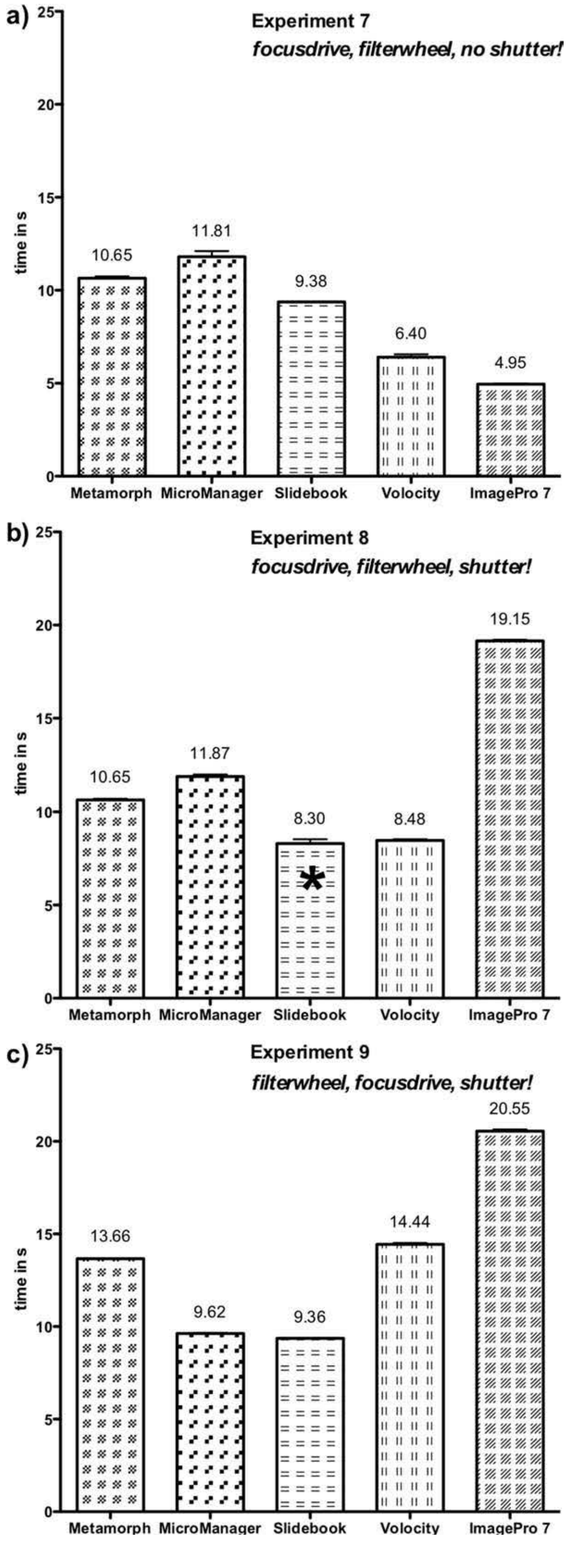

Fig. 5. Performance time comparison for a two channel *z*-stack experiment with 10 slices. (**a**) Experiment 7: fast acquisition performance of the five programs tested, without shutter closing between frame acquisition. First *z*-stack for Channel 1, then *z*-stack for Channel 2. (**b**) Experiment 8: acquisition performance with standard settings including shutter closing. First *z*-stack for Channel 1, then *z*-stack for Channel 2. *Acquisition time for Slidebook was clocked because no time stamp was available. (**c**) Experiment 9: acquisition performance with standard settings including shutter closing but first channel, then z-stack acquisition.

In Experiment 7, ImagePro7 required 5 s, Volocity 6.4 s, SlideBook 9.4 s, MetaMorph 10.7 s, and Micromanager 11.8 s (Table 2). The respective times for Experiment 8 were: SlideBook 8.3 s, Volocity 8.5 s, MetaMorph 10.6 s, MicroManager 11.9 s, and ImagePro7 19.1 s (Table 2). For this experiment, we were unable to read the performance time for SlideBook from neither the time stamp nor log file. The acquisition time was instead measured using a stopwatch (start: clicking “start” in the Multi Dimension Acquisition dialog, stop: when the Multiple Acquisition Window was closed). For Experiment 9, the following performance times were calculated: SlideBook 9.4 s, MicroManager 9.6 s, MetaMorph 13.7 s, Volocity 14.4 s, and ImagePro7 20.6 s (Table 2).

DISCUSSION

In our article, we tested and compared the performance time of five different generic-control software applications for microscopes (Image-Pro Plus, MetaMorph, Micro-Manager, SlideBook, and Volocity) using typical experimental setups with a single specific state-of-the-art fluorescence microscope configuration.

We are aware that we have not fully optimized our hardware configuration from the speed point of view. Triggering the shutter directly through the camera, or preprogramming the filter-wheel and triggering it through either the camera or computer, would have certainly resulted in shorter acquisition times in most of the experiments. However, this would have limited the measurement flexibility, which in most cases is well appreciated by the users—so we decided to retain full software control. While the use of “real-time controllers” offers an elegant solution to this speed/experiment flexibility problem, these systems are bound to specific hardware components and the aim of this article is to compare software performance using identical hardware installations.

Furthermore, it is important to emphasize that software was self-installed by the authors, and the hardware control settings were implemented in accordance to software manuals, and after consultations with company hotlines. We ensured that all hardware components were properly managed by the controlling software.

To evaluate the performance time, we calculated the theoretical “minimal time” needed for each experimental setup from the denoted response times of the different hardware components. This occurred in accordance with the respective technical specifications (e.g., camera readout time 112 ms) (Table 2) under the assumption that the various hardware "steps" follow each other in a serial manner without time-delay between them.

By examining the overall performance times no "ultimate winner" emerges, meaning that none of the tested software can be considered as the fastest in all experimental setups. With the exception of the burst-acquisition, the actual system speed (due to the software) is suggested to be always a factor of 2–8× slower than what is theoretically possible. By comparing and analyzing the experimental performance of the various software applications, it is possible to clearly identify the "strengths" and "weaknesses" of the products. Considering hardware components separately, we can determine which applications control them in an optimum way and which show large deficiencies.

Camera

In Experiment 1 ("time lapse, burst acquisition") the camera was the only "extra" component controlled by the software. MetaMorph, MicroManager, and Volocity were very close to the calculated minimal time, whereas SlideBook and ImagePro7 were significantly slower, indicating that for these programs the camera readout could be optimized further.

Shutter

Experiment 2 ("time lapse with one channel and remote shutter") and subsequent even-numbered experiments give an idea of the shutter control speed of the various software applications. In this respect, SlideBook seems to slightly outperform all others. Also notable is: in a situation requiring control of the other nonshutter components (filter wheel or *Z*-drive), using a shutter has no effect anymore on the performance time. Exceptions are Volocity (small increase) and ImagePro. This latter application appears to have a significant problem with optimum control of the shutter. Interestingly, using the shutter can even slightly improve the performance time (MetaMorph, MicroManager), but it certainly increases its reproducibility. We currently have no explanation for this observation.

Filter-Wheel

The filter-wheel speed measurements (Experiments 3 and 4) show the largest relative deviation (~4× in the fast case) from the anticipated value. This indicates substantial room for improvement, but also highlights the possibility of incorrect values given by the hardware manufacturer. Furthermore, it is striking that there is additionally almost a 2× difference between the fastest (ImagePro) and the slowest (MetaMorph) programs. It is additionally clear that filter wheels are one of the major speed limiting factors in fast acquisition. The situation would have been considerably worse, had the motorized filter turret of the microscope been employed. This is approximately a factor 5 slower then the filter-wheel. Innovative solutions from the component manufacturer side would additionally be needed in this case.

Z-Drive (Microscope Control)

From the theoretical (hardware) point of view, the slowest component is the *Z*-drive (microscope). It is still remarkable that both SlideBook and ImagePro can control the *Z*-drive faster than the specification given by the manufacturer (Experiment 5). This shows that, at least in some cases, optimum control can be reached. Nevertheless, there is still a significant 2× difference observable between the fastest and the slowest (SlideBook and MetaMorph) programs.

Multiplexed Experiments

Measurements 7, 8, and 9 are essentially a combination of Experiments 3, 4, 5, and 6. Accordingly, the corresponding results can be partially understood based on the underlying measurements. However, some interesting observations can be still noted. In line with the single component measurements, ImagePro seems to be the fastest software when shutter-control is not required. However, this "trend" is contradicted by the relationships between MetaMorph and MicroManager, and between Volocity and SlideBook. In both cases, based upon the single component measurements in Experiments 7, 8, and 9, the latter software applications (MicroManager and SlideBook) were expected to be faster. However, this is only true in Experiment 9. A possible explanation is that the run speed of a complex experimental workflow is to a large extent also dependent on the control synchronization of the individual components. It is presumed that MetaMorph and Volocity perform better in this regard, and this is why they can achieve a better performance time, even though they are slower at individual control of the peripheries. In Experiment 9, involving many control steps of the filter-wheel, MicroManager and SlideBook benefit from their speed advantage when controlling the filter wheel. They show an overall improved performance time despite their obviously suboptimal component synchronization.

SUMMARY OF THE SOFTWARE-SPECIFIC OBSERVATIONS

MetaMorph

Clearly the fastest software when only the camera needs to be controlled, but less optimal for the shutter and filter-wheel. The main weakness is certainly the microscope *Z*-drive control. A possible explanation is that MetaMorph is the only software controlling the microscope through the Zeiss MicroToolBox interface provided by Zeiss.

MicroManager

The only open source (free) product on our test performed remarkably well, especially considering the absence of the development resources associated with a commercial product. The major weakness is the microscope (*Z*-drive) control.

SlideBook

Most-consistent performance. Remarkably fast in controlling all peripheries, but camera-control speed could still be improved. It is also remarkable that the repeated SlideBook time-measurements showed the smallest variations, additionally indicating a very reliable and reproducible control performance.

Volocity

Relatively consistent and fast performance but shutter control could certainly be improved.

ImagePro

Best control performance for the filter wheel and the microscope (*Z*-drive), but camera control could be improved. Shutter control certainly needs improvement, as the primary cause of poor performance in most of the tests.

CONCLUSIONS

Today, there is a large choice available of software to control wide-field microscope systems. Although the market is likely still growing, given the increasing importance of imaging in biology, it is already a highly competitive business. Market players (including also the microscope manufacturers) increase the attractiveness of their software by including increasingly sophisticated analysis features. Unfortunately, in the process, it is partially forgotten that these software applications were originally and primarily made for acquisition control. Our study clearly demonstrates much potential for optimization with all the tested software applications. The study also suggests that only comparative studies of this nature can pin point the weaknesses of the various products. Additionally, it is also evident that existing direct-control methods from a computer cannot make use of the full speed of the system. More sophisticated solutions which offer both control flexibility and maximum speed are desirable. Some companies have already made efforts in this direction by including real time controllers into their system setups, but they are far from generic and can control only a very limited number of peripheries.

ACKNOWLEDGMENTS

The authors thank PerkinElmer (Volocity) and Media Cybernetics (Image-Pro Plus) for providing us with free licenses for software testing, and Owen Feehan for revising our manuscript.

REFERENCES

Murray JM. 1998. Evaluating the performance of fluorescence microscopes. J Microsc 191:128–134.

Murray JM, Appleton PL, Swedlow JR, Waters JC. 2007. Evaluating performance in three-dimensional fluorescence microscopy. J Microsc 228(Part 3):390–405.

Swedlow JR, Hu K, Andrews PD, Roos DS, Murray JM. 2002. Measuring tubulin content in Toxoplasma gondii: A comparison of laser-scanning confocal and wide-field fluorescence microscopy. Proc Natl Acad Sci USA 99:2014–2019.

Vermont J, Fraser SE, Liebling M. 2008. Fast fluorescence microscopy for imaging the dynamics of embryonic development. Hfsp J 2:143–155.

Wang E, Babbey CM, Dunn KW. 2005. Performance comparison between the high-speed Yokogawa spinning disc confocal system and single-point scanning confocal systems. J Microsc 218(Part 2):148–159.

Zucker RM, Price O. 2001. Evaluation of confocal microscopy system performance. Cytometry 44:273–294.