



HAL
open science

Computer-aided drawing system – substitute for camera lucida

E.A. Sidorchuk, D.D. Vorontsov

► **To cite this version:**

E.A. Sidorchuk, D.D. Vorontsov. Computer-aided drawing system – substitute for camera lucida. *Acarologia*, 2014, 54 (2), pp.229 - 239. <10.1051/acarologia/20142130>. <hal-01565290>

HAL Id: hal-01565290

<https://hal.science/hal-01565290v1>

Submitted on 19 Jul 2017

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



Distributed under a Creative Commons CC BY-NC-ND 4.0 - Attribution - Non-commercial use - No Derivative Works - International License



ACAROLOGIA

A quarterly journal of acarology, since 1959
Publishing on all aspects of the Acari

All information:



<http://www1.montpellier.inra.fr/CBGP/acarologia/>
acarologia@supagro.inra.fr



**Acarologia is proudly non-profit,
with no page charges and free open access**

Please help us maintain this system by
encouraging your institutes to subscribe to the print version of the journal
and by sending us your high quality research on the Acari.

Subscriptions: Year 2017 (Volume 57): 380 €

<http://www1.montpellier.inra.fr/CBGP/acarologia/subscribe.php>

Previous volumes (2010-2015): 250 € / year (4 issues)

Acarologia, CBGP, CS 30016, 34988 MONTFERRIER-sur-LEZ Cedex, France

The digitalization of Acarologia papers prior to 2000 was supported by Agropolis Fondation under the reference ID 1500-024 through the « Investissements d'avenir » programme (Labex Agro: ANR-10-LABX-0001-01)



Acarologia is under **free license** and distributed under the terms of the Creative Commons-BY-NC-ND which permits unrestricted non-commercial use, distribution, and reproduction in any medium, provided the original author and source are credited.

COMPUTER-AIDED DRAWING SYSTEM – SUBSTITUTE FOR CAMERA LUCIDA

Ekaterina A. SIDORCHUK¹ and Dmitry D. VORONTSOV²

(Received 09 March 2014; accepted 16 May 2014; published online 30 June 2014)

¹ *Arthropoda Laboratory, Paleontological Institute, Russian Academy of Sciences, Moscow, Russia. e.a.sidorchuk@gmail.com*

² *Laboratory of Comparative Physiology, Institute of Developmental Biology, Russian Academy of Sciences, Moscow, Russia. colupaika@gmail.com*

ABSTRACT — The drawing system we describe consists of a microscope, digital video camera, computer with a mouse or a pen tablet, and freely available software. The system uses an overlay technology that allows drawing with either raster or vector software, using a live image background that is translated either from the microscope via a digital camera or from previously captured images. With some practice, this system allows one to make high-quality illustrations directly from a microscope, omitting the time-consuming steps of drawing and inking pencil sketches or taking, combining and importing photo-stacks to drawing software for subsequent outlining.

KEYWORDS — methods; techniques; scientific illustration; morphology

INTRODUCTION

With the development of computer drawing software, the most time-consuming part of taxonomic description — preparation of drawings — became much optimized by allowing either the "inking" of pencil sketches after scanning them, or the outlining of necessary details from a digital photograph or image stack (Coleman, 2006; Fisher and Dowling, 2010; Tereshkin, 2008). The second method speeds up the drawing process by excluding the time-consuming stage of a pencil sketch and allows production of a high-quality drawing suitable for publication without further adjustments.

This second method, however, does include another time-consuming stage — combining the stack of photographs and/or importing multiple images to the drawing software — if the object is three-dimensional and requires multiple images to represent it properly. Also, digital photographs taken

from the microscope do contain artifacts. Those can become misleading in the images, received by combining captures of subsequent focal planes, especially if it is done automatically. Direct observation under the microscope thus remains necessary to avoid misinterpretation of complex structures. The last but not the least problem is to choose an appropriate magnification and resolution to capture both the outline and the fine detail necessary for taxonomic description. While not so important for large arthropods, this problem stands out sharply when describing small arthropods, such as mites. This is probably why many mite taxonomists still prefer making pencil sketches with the aid of a drawing-tube, or *camera lucida*. The main advantage of this optical device (Hammond and Austin, 1987) over using a stack of images is the direct microscopic observation of the specimen that is being drawn. One can adjust illumination, contrast methods and focal point of the microscope while drawing, to bring

into focus a detail of interest, even if it is newly discovered or minute, while photographs must have captured all the important details at the very beginning. This becomes especially difficult when documenting inclusions in amber, where details emerge or become obscure depending on the angle of illumination.

This contribution shows how one can combine the advantages both of the computer-aided drawing and of the *camera lucida*. The method described below allows making readily printable digital drawings (see figures in Sidorchuk and Norton, 2010; Klimov and Sidorchuk, 2011; Sidorchuk and Klimov, 2011; Sidorchuk and Norton, 2011a, b; Sidorchuk and Bertrand, 2013; Sidorchuk *et al.*, 2014; Norton and Sidorchuk, in press.) from a live image acquired directly from the microscope, supported by direct observation through the eyepieces. The system is cost-effective, since it is component-flexible and uses the full power of literally any drawing software, including freeware and open source.

It is important to note, however, that newer versions of software are not always better than the older ones, especially when it comes to the functions which are not very commonly used. Therefore, the first installation of the drawing system we describe can be difficult in some operating systems (in Windows 8, for example). Also, using the appropriate version of VLC media player is essential, although we do not lose hope that the functionality required for our method will be restored in newer versions of this media player or implemented in some other softwares. We describe the method as we apply it (in Windows XP and Windows 7), providing all the necessary detail, and encourage discussion and application of the other software that we did not try.

MATERIALS AND METHODS

The system we propose consists of a microscope, a digital camera attached to it, and a computer with a mouse or (preferably) with a pen tablet. The image from the camera is translated in a real-time mode to the computer screen where it becomes the back-

ground for the drawing software. The illustrator outlines the object using the tools of the drawing software while adjusting the microscope and position of the object as necessary. Since the live image is established as a background, the drawing procedure is essentially the same as if using a stack of images (Coleman, 2006), but with an important advantage: to change focal point or illumination, one needs only to adjust the microscope, rather than having to take additional images or cycle through layers in a stack.

The initial and crucial step of this method is to establish the live image from the microscope as a background for drawing. This can be accomplished using a so-called 'overlay mode' for showing a video image on the computer screen. Overlay is widely used in different software, such as media players, to reduce central processor load while transferring a video stream to the monitor. In overlay mode the software does not draw every video frame on a screen pixel by pixel. Instead, it fills some part of the screen where the video has to be shown with a specific overlay color (usually dark-blue, almost black) while all the rest is done by the hardware video processor, which finds and replaces overlay color with the picture from a video file or from the attached video camera.

To establish a live image from the camera in the background of the drawing software, one needs to (i) display the live (video) image in overlay mode, and (ii) fill the background of the drawing with that specific overlay color. The second part of the task can be achieved in almost any drawing software, either vector or raster. To display the image, one needs a video camera attached to the microscope and the software capable of receiving the video stream from the camera and showing it immediately on a screen in overlay mode. We have tested a few programs capable of performing this task and have chosen only one for our work, the open source VLC media player, but there may be others. Also, while our method was developed in MS Windows XP and MS Windows 7 operating systems, it should be applicable to others, such as Linux or Mac OS X. Unfortunately, Windows 8 seems to be incompatible with the overlay mode realization in older ver-



FIGURE 1: An example of a working place with the complete drawing apparatus ready to use. C-camera, T-pen tablet

sions of Windows, so that we cannot recommend it for this application. The method is described in detail below, along with the particular set of hardware and software that we have chosen for our own drawings. **Bold text** indicates the crucial steps of the installation process.

Workspace and setup — While drawing, it is convenient to have easy access to controls and eyepieces of the microscope, a computer monitor within view that produces minimum eyestrain, and input tools (keyboard, pen tablet or mouse) positioned for access without unnecessary reaching. Our workspace (organized in space-restricted conditions) is shown in Fig. 1.

Microscope — The microscope must have all the optical capabilities one normally requires for observation. A trinocular head is preferable, as it leaves both eyepieces available while the camera is in use. For close examination of mite inclusions in amber magnifications of 100-1000 \times are needed, so we use a Jenaval compound microscope with both normal transmitted illumination and epi-illumination from an external fiber-optic light source.

Digital camera — The best choice is a camera designed for use with a microscope. If a professional camera of this kind (*i.e.* made by Leica, Zeiss, etc.) is too expensive, one can successfully use one

of the cheaper USB cameras that recently have become available. If the computer runs the Windows XP or Windows 7 operating system, as in our case, the main requirement is that the camera driver supports the DirectShow standard. The first camera we worked with was a ScopeTec DCM 500, with a maximum resolution of 2592 \times 1944 pixels (px) and maximum capture-rate of 30 frames per second (fps). Half that resolution and frame rate is also acceptable. The other cameras we successfully used are AmScope MU800 and AmScope MU900.

Computer — Virtually, any modern computer is appropriate for the task (however, see our comments concerning the operating system). The faster its processor and the larger its operative memory (RAM), the smoother the whole drawing system will perform and the higher-resolution drawings can be made.

We have successfully used the following computers:

- (i) Dell Inspiron 600m laptop, (Intel Pentium M processor at 1.6 GHz, 512 Mb RAM, ATI Mobility Radeon 9000 video card) which had worked slowly with raster drawings of 6,000 \times 6,000 pixels in size and video stream format 1280px at 15 fps.
- (ii) desktop PC (Intel Core 2 Duo processor E7500 2.93 GHz, 4 GB RAM, ATI Radeon HD 4550 512M

video-card) which allowed comfortable working with drawings of up to 10,000×10,000 px receiving from the microscope a video stream of 2,592×1,944 px at 30 fps.

(iii) Samsung NP900X4C laptop (Intel Core i7 processor 3517U 1.9 Ghz, 8 Gb RAM, Intel HD Graphics 4000).

Pen tablet — Though one can draw using a mouse or a touchpad, a pen tablet has proven essential for efficiently making good drawings. The better the pen tablet, the easier the process will be, so if the budget is restricted, we recommend saving on other components, not on this one. The necessary features of the pen tablet are: (i) a cordless and preferably battery-free input tool, (ii) pressure sensitivity (this allows easy drawing of natural-looking lines), (iii) high resolution, and (iv) high accuracy. We use a Wacom Intuos 4M tablet with a pen-stylus (working area 223,5×139,7 mm, coordinate resolution 200 lines per mm, accuracy of the pen ± 0,25 mm).

Software

Operating system — We use Windows XP SP2 or Windows 7 (<http://www.microsoft.com/>) on our computers. As mentioned above, it should be possible to realize the same method under any operating system that supports overlay technology. Windows 8 does not fully support the overlay technology and we cannot recommend it for this application.

Video software — Video software has to support an overlay mode and be capable of performing live capture from the video camera. It is very convenient for drawing when the software also has a "video on a desktop" mode. If the software supplied with the camera supports an overlay mode and can display the live image on a desktop, no additional video software is required. Software supplied with our cameras could not work in overlay mode, so we use an open source VLC media player ver. 0.9.8a. This is not the latest version of this media player, but the old versions, including the one we use, are available: http://www.oldapps.com/VLC_Player.php, <http://download.videolan.org/pub/videolan/vlc/>. Newer versions of this media player exist, but they perform worse for our application, as often happens

with rarely used properties of programs. Obtaining the appropriate, old version of VLC media player specified above is necessary to follow our protocol, although we hope that, if set up properly, other software would also work.

Setting up VLC media player and obtaining overlay color — **Download the VLC media player ver. 0.9.8a and install it. Run VLC player and disable its automatic updates.** The media player can use or not use the overlay mode, displaying the same image in both cases, while only the overlay mode is suitable for our needs. **The preferences of VLC media player must be set as in Fig. 2** to proceed. From VLC player menu, open 'Tools > Preferences > Video' in order to see the 'General Video Settings' dialog (Fig. 2). In the 'Output' dropdown menu choose the 'DirectX video output'. Also check the 'Enable wallpaper mode' option (arrows in Fig. 2). **Save settings.** Close VLC media player and open it anew to make sure that the saved settings are applied. In Windows 7 and Windows Vista, it is safer to disable the 'Aero' interface: choose 'Start > Control Panel > Personalization' and select any of the themes in the 'Basic And High Contrast Themes' category.

To check whether the player is utilizing the overlay mode, open some video file ('Media > Open file'). A still image file may also be used instead of a video file (this reliably works in Windows XP, but not in Windows 7), as shown in Fig. 3A. Make a screenshot (press a 'Print Screen' key on a keyboard). Run the drawing software and paste the screenshot image from the clipboard and make the pasted replica of the media player window overlap the original window of the media player (Fig. 3B). Part of the image translated by the media player should emerge through the replica. It must be noted that this effect may depend on the magnification of the image (zooming) in the window of the drawing software: for example, in GIMP the area filled with overlay color looks invariably black being scaled to 66.7% but behaves as in Fig. 3B at the other magnifications.

Close (not minimize) the media player. The contents of the replica of media player window in the screenshot should become dark if the overlay mode

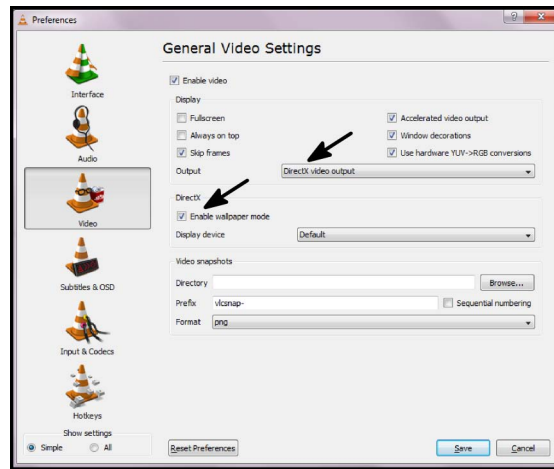


FIGURE 2: Settings of VLC media player. Those important for correct and convenient work in overlay mode are indicated by arrows. This dialog can be found under the 'Tools > Preferences' menu.

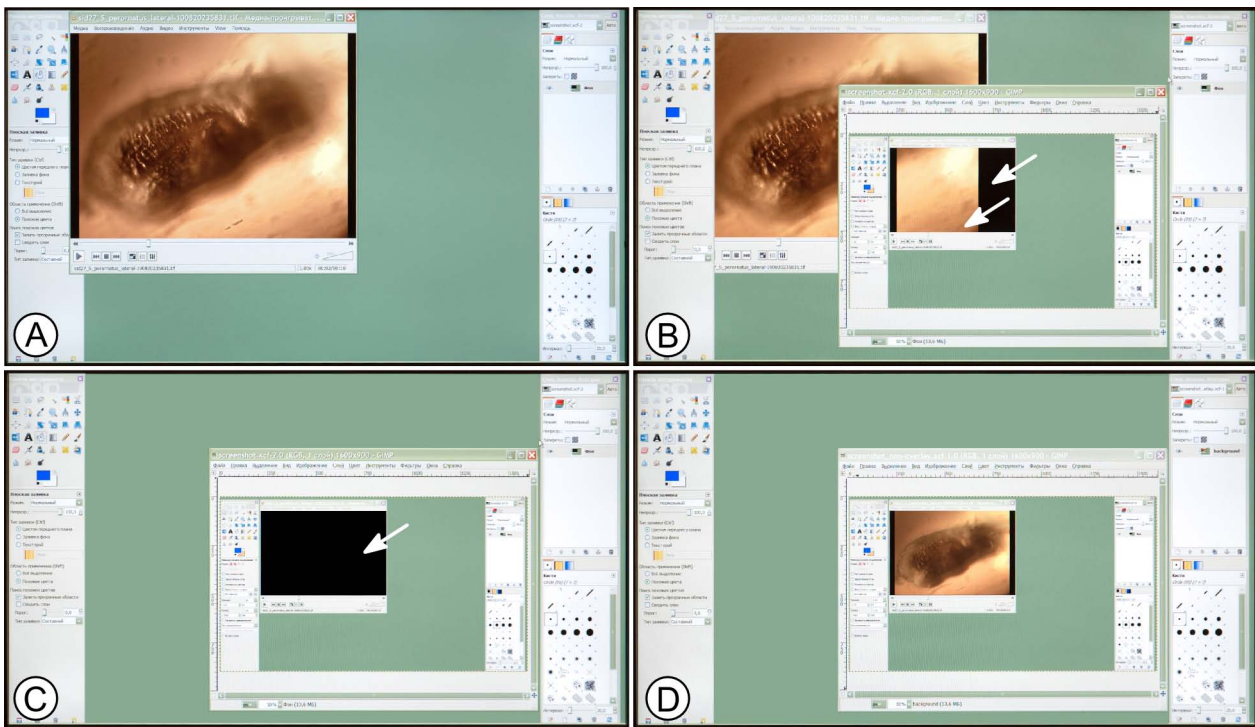


FIGURE 3: Setting a live image from the camera as the background for drawing (software used for this example are VLC media player and GIMP graphic software). Since the overlay mode image cannot be reproduced in a screenshot, we photographed the computer screen using a digital camera instead of making the combined images from incomplete screenshots: A – media player showing the image; B – screenshot taken from A and pasted to the new file in the drawing software. The overlay mode is active: part of the video frame is visible through a replica of the media player's window where it overlaps with an original window of the media player, the rest of the replica is dark but is nevertheless filled with overlay color. Both arrows point to the area filled with overlay color; C – media player is closed, the whole area filled with overlay color looks dark; D – same as C, but the media player was *not* in overlay mode when the screenshot was taken. There is no overlay color on the screen. See text for further explanations.

was active (Fig. 3C). If it still contains the image which was demonstrated by the media player, then the media player did not use the overlay mode (Fig. 3D). This can be due to any of the following reasons: the VLC player settings are wrong, its version is different from the one we recommend (you can check the version under 'Help > About' menu), or the operating system does not support the overlay mode. If everything is set up correctly, the player should use the overlay mode. If it nevertheless does not use overlay mode, try stopping VLC player via the 'Task manager' and running it anew, or restart the computer and try again.

Pick up the overlay color (Fig. 3B, arrows) with a 'pipette' tool and **save it** for further use as a background for drawing from the live video streams, still captures, or video files. Normally, overlay color in MS Windows is RGB 0:0:1. However, in InkScape under Windows XP and Windows 7, there is a bug in setting the background color: one needs to set RGB color to 0:0:2 and transparency (Alpha channel) to 255 to get the right background color, although 0:0:1 is still good for drawing shapes. In some other software, the same setting can be slightly different, and it is worth trying different options if you surely have the live image from your microscope in overlay but cannot get it in the background of your drawing.

Driver of the video camera — **Plug in the video camera and install it according to the manufacturer's instructions.** The availability of the video camera in the VLC media player's list of capture devices (Fig. 4B) indicates that the correct video camera driver is installed. Three video cameras we used performed differently in this respect. In Windows XP, both ScopeTec DCM 500 and AmScope MU800 did not create any problems, same was true of the ScopeTec DCM 500 in Windows 7. Once a camera was plugged in and its software was installed from the CD supplied with it, the camera became accessible through VLC media player, as described in the next section.

AmScope MU900 camera needed additional steps to make it work with VLC media player both in Windows XP and in Windows 7: after the camera's software installation the camera did work with

that software (ToupView) but did not appear in the list of capture devices of VLC media player. The reason was that not all required drivers were installed during the software installation, although the drivers were present on the same disk supplied with the camera. After they were manually installed, the video camera became available through VLC media player (Fig. 4B).

Establishing the live capture — To establish the live capture from the camera through VLC media player, one has to **learn the camera's resolution closest to the resolution of the computer's screen.** Higher resolution would require more memory but would not be useful as the display would not be capable of showing it; lower resolution is resource-saving but can lead to the loss of detail. It is expedient to adjust the driver properties to obtain live capture of the best quality within the display resolution that permits the computer to show it in a real-time mode at about 15-20 fps or more. Setting a live capture resolution higher than that of the display is worthwhile only when one is drawing an enlarged part, not the whole image.

The currently used resolution of the display in Windows 7 can be learned by going to 'Control Panel > Appearance and Personalization > Personalization > Display Settings'.

Available resolution modes of the camera are most easily found through its specific software: menu 'Video stream format' in the ScopePhoto (the software specific for the ScopeTec cameras) and 'Capture&Resolution' in the ToupView (the same for AmScope cameras). For use with VLC media player, one has to record the larger parameter which is usually the horizontal size of the frame, in pixels. For example, AMScope MU900 has the following resolution (frame size) options: 3488 x2616, 1744x1308 and 872x654 px. For the screen with resolution of 1600x900 px, the middle option is expedient, so that one should set the 'Video size' parameter of VLC media player equal to 1744 (see below).

To **establish the live video stream from the camera through VLC media player**, open 'Media' menu and click 'Open Capture Device' (Fig. 4A). Ensure that 'Capture mode' is set to 'DirectShow'. From the list of available video devices, choose your

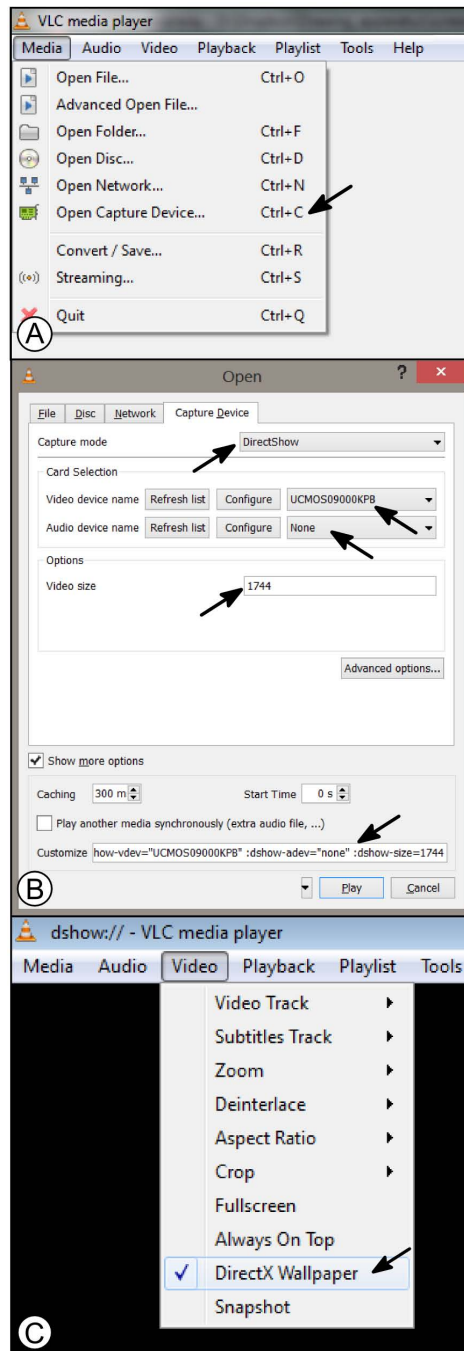


FIGURE 4: The procedure of setting the video camera as a source of live capture image in VLC media player: A – click ‘Open capture device’ under ‘Media’ menu; B – choose the video camera (‘Video device name’), set ‘Audio device name’ to ‘None’ and set the horizontal pixel size of the video image — it must be chosen from the predefined list specific for particular video camera or, alternatively, left blank. In the latter case the video, most probably, will have the lowest possible resolution. The bottom arrow points to the command line string which is generated by VLC media player and which can be used to automate the procedure of setting the video camera parameters; this command line is *not* required for the method described in this paper; C – setting the live capture image as a desktop background for convenient drawing.

video camera; set the audio device to 'None' and set the preferred video size (Fig. 4B, arrows). Click 'Play' button. An image from the camera should appear in the media player's window, maybe after some delay. The error message 'VLC cannot use the device "none"...' is related to the sound and should be ignored.

The procedure of choosing the proper video camera and its resolution can be (optionally) automated in VLC media player. To do that one needs to obtain the command line string that includes all the required settings. This string is provided by VLC player when the 'Show more options' checkbox is checked in the bottom of the 'Open' dialog (Fig. 4B). When all parameters in the upper part of this dialog are set properly, one can copy the command line string (Fig. 4B, bottom arrow) and create the desktop shortcut for VLC media player with the command line parameters added.

For convenience of drawing, we recommend to check the 'DirectX Wallpaper' option under the 'Video' menu of VLC media player (Fig. 4C)—this allows having the live capture image expanded over the larger part of the screen. This option also allows having the live image always scaled in the same way, which is very convenient (see 'Scale bars' section).

Camera settings — The video camera driver allows **setting the resolution, color options, frame rate and the other parameters**. Usually it has a separate dialog window, which can be opened from VLC media player. Restart the media player, choose your camera (see Fig. 4), click 'Configure' button near the video device name, adjust settings as needed, save them and click 'Play' button. In some cases due to the bug in the camera's driver it does not allow adjusting and saving the camera's parameters from anywhere except the camera's specific software — as we experienced with the ScopeTec camera. In this case all the settings were made from the ScopePhoto software before opening VLC media player.

If color is not required, grayscale mode is expedient. In our setups, VLC media player represents colors from the digital camera incorrectly, but this is not a concern in grayscale mode, which is also less

eye straining. The image may need to be flipped (for all of our video cameras it can be done in camera settings), so that movements of the image on the screen are in accordance with those in the eyepieces when the microscope stage is moved.

Drawing software and establishing a live image as a background to the drawing — Whether raster or vector, the drawing software should provide full support for the chosen pen tablet. Our present choices are GIMP (<http://www.gimp.org/>) for raster graphics and InkScape (<http://www.inkscape.org/>) for vector graphics, both of which are open source. **Set the media player to show video on the desktop in overlay mode** or, if this option is not expedient, maximize the video window and put the drawing software window on top of the media player window. **Change the background color of the drawing to overlay color**. If this is for some reason impossible, draw a rectangle filled with overlay color. As soon as it is done, the image displayed by the media player emerges through the window of the drawing software and can serve as a background for the drawing.

When all software is set up and working, **create a new transparent layer and draw within it**. The procedure of drawing does not differ much from that with a still image, and we refer reader to the protocols published by Coleman (2006), Fisher and Dowling (2010), Tereshkin (2008) and references there in. The difference is that one can draw while adjusting the focal point of the microscope, illumination and position of the object, scaling and moving the object as necessary (see Results and Discussion). As long as the color used for drawing differs from the overlay color, the drawing shows on the top of a live image. At this stage we suggest the use of a bright color (green or blue) that contrasts with the microscopic image. Later, it can be replaced it with a more conventional black. A white background layer can be created and mostly kept invisible. Making it visible from time to time helps evaluating the drawing while it develops.

RESULTS AND DISCUSSION

With practice, the drawing system described here allows making printable drawings directly from the live-captured image of the microscopic object (for examples of raster drawings, see figures in Sidorchuk and Norton, 2010; Klimov and Sidorchuk, 2011; Sidorchuk and Klimov, 2011; Sidorchuk and Norton, 2011a, b; for examples of vector drawings see Sidorchuk and Bertrand, 2013; Sidorchuk *et al.*, 2014; Norton and Sidorchuk, in press.). However, certain mistakes can cause a need for additional work on the drawings, or in some cases even totally remaking them to create printable figures (*i.e.* appropriate for submission with a journal paper). Our initial mistakes concerned drawing resolution, line width, line smoothing and scale bars. The first and third mainly relate to raster graphics, but the second and especially the last can become irremediable even when working with vectors graphics.

Drawing resolution — While creating a new blank image for future drawing (especially a raster image), one has to keep in mind its final size and resolution. Journals commonly demand line drawings in their final size to be high-resolution (600 or 800 dpi) and fit a specified width, which varies among journals between 72 – 85 mm (one column) and 160 – 170 mm (two columns), and to not exceed a height of 225 – 235 mm. Thus, the canvas size should not be less than 5300×7000 px. If the computer doesn't allow comfortable working with such large images, two or four slightly overlapping partial drawings can be made and subsequently combined. The latter may be necessary when working with a mouse, touchpad or inaccurate pen tablet in a raster software: to reduce line irregularities, one is obliged to zoom an image out to 50 % or even 25 % instead of 100 %, and increase line width and image size proportionally, with subsequent reduction of the image size.

Line width — It is very tempting to choose thin lines when drawing from the live capture image, because one sees every contour as a very narrow boundary, and magnification is usually rather high, so each detail seems to be large. Unfortunately, after zooming down to print preview, one discovers

an unhappy truth: outlines are thin and most of the carefully outlined details are obscure, while shadings are totally invisible. In vector graphics, this problem can be rather quickly fixed by increasing outline thickness, but even then line intersections can become misleading and fine detail may require simplification. In raster graphics line thickness is difficult to correct and it is not possible to thicken different lines to different extent. To avoid such problems, the line width for the main outline of an image fitting a 5300×7000 px canvas should not be less than 30 px, and the thinnest line on the drawing (or dots used to designate shadings) should not be less than 6 px. The same is true of dotted or dashed lines: spaces between dots and dashes cannot be less than the line width, otherwise they blend together as scale is reduced. In a raster file, to check if the drawing is printable, one can zoom in to see it at about 300 % of the print size (lines should be almost smooth) and zoom out to about 30 % of the print size to see if all the lines and shades are readily visible. If so, the drawing is printable. Zooming to about 30 % is also helpful for evaluating a vector drawing.

Line smoothing — This function should be avoided while working with raster images, as line drawings are usually printed in black and white, not grayscale. When transferred from full-color or grayscale, smoothed lines can become "shaggy" instead of retaining their original smoothness, or even disappear (see example figures in Sidorchuk and Vorontsov 2010).

Shifting and zooming — It often happens that with desired magnification, an object is larger than the screen or working area of the drawing software. This is not a problem, as the digital camera usually utilizes only the central part of a microscope's field of view, so that distortions at the edges of drawing are minimal. Thus, as soon as the observable area is drawn, one can shift a microscope's stage for the neighboring area of object to come in view, and drag the drawing to align it with the new position of the object. If magnification must be changed, one has to zoom in or out accordingly. To avoid problems with scale bars (see below), it is convenient to develop a table of magnification and zoom, an example of

TABLE 1: Accordance of magnification of a microscope and zoom of the drawing software.

objective	intermediate lens	zoom in Inkscape (vector)	zoom in Gimp (raster)
3.2X	0.8X	25.6%	1%
12.5X	0.8X	100%	4%
25X	0.8X	200%	8%
40X	0.8X	320%	12.8%
100X	0.8X	800%	32%
3.2X	1.0X	32%	1.3%
12.5X	1.0X	125%	5%
25X	1.0X	250%	10%
40X	1.0X	400%	16%
100X	1.0X	1000%	40%
3.2X	1.25X	40%	2%
12.5X	1.25X	156.3%	8%
25X	1.25X	312.5%	12.5%
40X	1.25X	500%	26%
100X	1.25X	1250%	51%

which is given in Table 1.

Scale bars — As the microscope magnifications are usually stepwise (*i.e.* objectives $\times 12.5$, $\times 25$, $\times 40$ and $\times 100$), it might seem useful to make a set of corresponding scale bars. This is only worthwhile if: one always works with the same computer screen of a given pixel size; resolution of the camera is chosen once and does not change; video is always translated to the desktop with the same scale factor; and changing magnification of the microscope is always accompanied with zooming the drawing accordingly. Each of these parameters affects the actual scale of the video as it is translated into the background of the drawing software. Unless these parameters are all set in a routine manner, to assure the correct scale in every case, it is better to use the following procedure. When the drawing is finished or has to be set aside, without changing anything (zoom settings, magnification of the microscope, resolution or orientation of the camera), substitute the preparation with a stage micrometer and draw a single line in a new graphic layer with a text label indicating its actual length. Later, when labeling the figure and preparing it for submission,

the line may be rotated and overdrawn by a standard scale bar according to the journal's style.

Raster or vector? — Raster drawing most resembles normal, non-electronic, drawing: once colored, each pixel is completely like any other, and it is easy to erase lines without minding the types or interactions of objects. This makes easier the switch from sketching with the help of a *camera lucida* to sketching or drawing with the help of a computer. When drawing needs to be full-color, with shadings and subtle tint changes (see Tereshkin 2008 for multiple examples), raster graphic software provides major advantages, as it is primarily dedicated to image creation and processing. For simpler organized line drawings, however, vector graphic suites allow greater freedom in changing line widths and styles, scaling, and copy-editing (examples given by Fisher and Dowling 2010, their Figs. 5-7). Raster drawings tend to look more "natural", as raster graphics encourage the freehand outlining of every feature without trying to interpret it. Vector graphics encourage clarity and the limitation of minor detail, sometimes even oversimplification and the temptation to copy-paste some elements that are not truly

alike in the specimen, which can lead to inaccuracy. Although it may seem easy to convert the drawing from raster to vector and back, the resolution is not improved during the conversion: this can become problematic when the final illustration needs to be enlarged. Considering all these points may help in choosing the most appropriate software for the intended purpose.


ACKNOWLEDGEMENTS

We are grateful to Prof. Roy A. Norton (State University of New York College of Environmental Science and Forestry, USA) for valuable suggestions and improvements to an earlier version of this manuscript. We cordially thank Dr. Irina D. Sukacheva, Dr. Dmitry V. Vasilenko, Prof. Alexandr P. Rasnitsyn and Prof. Alexandr G. Ponomarenko (Borissyak Paleontological Institute, Russian Academy of Sciences, Moscow) for their continuous encouragement and suggestions. We are thankful to the two anonymous reviewers for constructive criticism of our manuscript and to Dr. Evert E. Lindquist (Canadian National Collection of Insects, Arachnids and Nematodes) for comments and suggestion to its revised version. The Russian Foundation for Basic Research provided financial support to EAS (grant RFBR # 12-04-01177-a).

REFERENCES

- Coleman C.O. 2006 — Substituting time-consuming pencil drawings in arthropod taxonomy using stacks of digital photographs — *Zootaxa*, 1360, 61-68.
- Fisher J.R., Dowling A.P.G. 2010 — Modern methods and technology for doing classical taxonomy — *Acarologia*, 50(3), 395-409.
- Hammond J.H., Austin J. 1987 — The camera lucida in art and science — Bristol: Adam Hilger, pp. xii+201
- Klimov P.B., Sidorchuk E.A. 2011 — An enigmatic lineage of mites from Baltic amber shows a unique, possibly female-controlled, mating — *Biological Journal of the Linnean Society*, 102, 661-668. doi:10.1111/j.1095-8312.2010.01595.x
- Norton R.A., Sidorchuk E.A. in press — *Collohmanna johnstoni* n. sp. (Acari, Oribatida) from West Virginia (U.S.A.), including description of ontogeny, setal variation, notes on biology and systematics of Collohmanniidae — *Acarologia*.
- Sidorchuk E.A., Norton R.A. 2010 — Redescription of the fossil oribatid mite *Scutoribates perornatus*, with implications for systematics of Unduloribatidae (Acari: Oribatida) — *Zootaxa*, 2666, 45-67.
- Sidorchuk E.A., Vorontsov D.D. 2010 — How to make a drawing apparatus with a microscope and computer [Internet] — [12 April 2014]. Moscow [01 July 2010]. Available from: (<http://www.openscience.ru/index.php?page=article&article=201>) [in Russian]
- Sidorchuk E.A., Klimov P.B. 2011 — Redescription of the mite *Glaesacarus rhombeus* (Koch & Berendt, 1854) from Baltic amber (Upper Eocene): evidence for female-controlled mating — *Journal of Systematic Palaeontology*, 9(2), 183-196. doi:10.1080/14772019.2011.566585
- Sidorchuk E.A., Norton R.A. 2011a — The fossil mite family Archaeorchestidae (Acari, Oribatida) I: redescription of *Strieremaeus illibatus* and synonymy of *Strieremaeus* with *Archaeorchestes* — *Zootaxa*, 2993, 34-58.
- Sidorchuk E.A., Norton R.A. 2011b — The fossil mite family Archaeorchestidae (Acari, Oribatida) II: redescription of *Platigeocranus sulcatus* and family-group relationships — *Zootaxa*, 3051, 14-40.
- Sidorchuk E., Bertrand M. 2013 — New fossil labidostomatids (Acari: Labidostomatidae) from Eocene amber and presence of an apustulate species in Europe — *Acarologia*, 53(1), 25-39.
- Sidorchuk E.A., Schmidt A.R., Ragazzi E., Roghi G., Lindquist E.E. 2014 — Plant-feeding mite diversity in Triassic amber (Acari: Tetrápodili) — *Journal of Systematic Palaeontology*, preprint, 1-23.
- Tereshkin A.M. 2008 — Methodology of a scientific drawings preparation in entomology on example of ichneumon flies (Hymenoptera, Ichneumonidae) — *Eurasian Entomological Journal*, 7(1), 1-9 + I-VII.

COPYRIGHT

 Sidorchuk E.A. and Vorontsov D.D. *Acarologia* is under free license. This open-access article is distributed under the terms of the Creative Commons-BY-NC-ND which permits unrestricted non-commercial use, distribution, and reproduction in any medium, provided the original author and source are credited.