Thu Anh Tran
Literature Review 2

Watercolor painting is prominent among the painting community but there exist challenges such as the existing paint applications are very bulky in size. The authors of “Painting with Polygons: A Procedural Water Engine,” Stephen DeVerdi, Aravin Krishnaswamy, Radomir Mech, and Daichi Ito proposed an algorithm that would provide the features of water painting that would allow dynamic paint behaviors and light weight. The size of the application is a very important issue if you want to reach users from all types of platform. For example, your application would gain massive attraction if it was accessible from PC, laptop, tablets, and mobile phone. The method that the authors used was a vector stroke representation and rendering of arbitrary resolutions so that it will be able to be used on slate devices. The researchers had developed an algorithm and built a productions paint application available on the digital marketplace, along with addressing the reviews by users. Lastly, the researchers compared their tool against the existing paint applications.

Although there have been large advances in digital painting, the digital world is unable to provide the wide range of complex textures and qualities to artists who uses water paint. The researchers have approached this challenge by providing a twist to it. Instead of the traditional water painting results, the authors have decided to add a dynamic range of behaviors that “inspired by the core watercolor effects such as blending, feathering, and edge darkening.” To do this, their algorithm uses particle-based model of pigment flow rather than grid based. Also, the particle is stored as vectors instead of raster. That allows for rendering in multiple resolutions. Lastly, the particle update functions is a procedural algorithm that reduces calculation time. The main features of their tool is providing users with a variety of different types of brush and pigment types to render common watercoloring results lie, edge darkening, nonuniform pigment density, granulation, and backruns.

As mentioned earlier, there are solutions on the market that provides watercolor painting capabilities. Although there still exist certain gaps. For commercial products such as

Corel Pater and Ambient Design ArtRage halts the processing routine unto an artist paints a brush stroke. By doing that, the user would have an interrupted workflow. There is nothing yet on the market that is able to capture the interactive and dynamic nature of watercolor painting. The nature of watercolor painting is very complex. Adobe Illustrator, and Ardeco system have tried to address that issue by using static vectors and turning arbitrary images into art decostyle vector representation. However, either technique is able to allow the user to control the brush stroke interface.

The algorithm these research developed tackles on the gaps that was not filled by the commercial tools. The logic behind their model is using a sparse representation of the paint pigment and not a dense representation which is normal what a normal pixel is compose of. Then the program will do a random walk algorithm to update the position of the pixel. That way, the approach is able to provide a more interesting behaviors of the pixel giving it the water color affect. The other steps include paint initialization, where each pain pigment is represented a dynamic “splat” particle. A splat is a complex polygon of n vertices. Then there is pigment advection, followed by sampling management, lifetime management and finally, brush type.

Using the techniques mentioned previously, the authors created an application that was tested, and used by the public crowd yielding a variation of review. Their paint applications took certain consideration into account in order to be workable with their proposed algorithm. The user interface is a desktop application that allowed the users to control the paint parameters, brush types, size, and changing the canvas texture. The user had a choice of outputting the result as a PNG or SVG file. Due to the small application size, it was made available to iPad applications unlike other paint applications mentioned earlier. The splat design made it work on low powered devices like the IPad. Users were polled for results and it include a mixture of different review scores. The users were impressive with the features that it provided but the application was not easy to use. Others say the interaction felt normal. It was a challenge for the users to analyze the result.

The conclusion was the authors developed a unique approach of a dynamic and interactive paint engines that produces many watercolor results. The commercial application they released was very popular with 4,500 downloads in the first day and 7th best-selling paid application.
Although with tremendous success there are still limitation that require future work to continue to explore the rage of effects to be able to be useful artist.

The second, more recent article titled, "Democratizing Digital Content Creation Using Mobile Devices with Inbuilt Sensors," written by Kapil Dev and Manfred Lau talks about using mobile devices to create contents. It is well known that mobile devices are growing rapidly, outnumbering stationary devices. That is why it is important to study this aspect and make digital creation possible via mobile devices. However, with being a mobile devices power supply, computational power, and display and interaction needs to be addressed. The researchers of this journal goal is to inform us about the current state of leading digital content creation via mobile devices with inbuilt sensors and address issues and future work for improvement.

There are many attributes of mobile devices that grew from and traditional desktops. For example, the ubiquity the mobile provides. In addition, the mobile devices allow the users to make a task easier. Mobile devices also come with sensors out of the box which provides many advantages. There is a large number of mobile content creation apps such as drawing, sketching, painting, 3D modeling, and object reconstruction.
The state of the art features of sensor-based content creation is broken down into 2D and 2D. In 2D content, sketching, drawing, and picture capture is taken into account. They are achieved by techniques that allow for direct and indirect automatic input, by the inbuilt sensors. Touch and accelerometers allows a mobile artist to manipulate the devices to achieve such result as realistic lighting effects or control shadowing. Dev and Lau mentioned that the touch and accelerometer feature of mobile devices was used by the application by Stephen Diverdi and his

colleagues. The detailed simulation of water color effects is resource intensive. However, Diverdi and team was able to build an app that uses lightweight geometric model to create brush strokes. The app creators also leverage the accelerometer to link its screen orientation with automatic gravity-based watercolor pigment flow. The success of that app supports that touch and accelerometer are important in digital content creation. Pen, touch, flash, and directional and positional sensors are also very important to content creation. The pen and touch give the users a natural realistic experience to draw or sketch on a mobile device. Flash allow users to create high quality videos. Directional and positional sensors are widely used in navigation application like Google maps that is accessible from any platform and application.

The 3D content creation requires more complex models, but the features provided with mobile devices make this possible. 3D object representation is constructed by polygonal and spine surfaces, sold geometry, and procedural modeling. The article written by Diverdi and team mentions that they use a procedural method to perform the rendering because it speeds up the calculation.

There are advances in mobile computing to allow digital content creation but also exists challenges to be addressed by more research. For example, there still exists limitation in the mobile devices but to the limitation of technology. The main constraint if in the hardware with emphasize on: computational requirements, sensing and input devices, and storage. The computational requirements talk about the increase realism of computer images, there is a positive correlation with the computation requirement of a mobile device. To address this challenge, the mobile computing technology have made advances in the architecture an organization of the mobile GPU. But the complexity of the digital images videos creates intensive tasks to the mobile device. The leaders have recommended that computationally intensive tasks to be moved to the remote servers so that the mobile devices only receive the result and not required to make any commutation that would take up resources and battery. For real-time collaborative work that we have seen more these days, the participating mobile devices can be programmed to use shared computing resources. Sensing and input devices are

naturally useful to be used to create digital content. As seen in the first article, Diverdi and his team created and app which allowed for user inputs. Sensing and input is very important in digital watercolor painting. For example, users are able to have access to such techniques such as creating washes, glazing, lifting off, , and unique features of watercolor medium. Stylus will allow users to control sensitivity to the object. There are also challenges in storage since 3D models require high computational and large capacity storage. Software written by engineers will also need to efficiently utilize the mobile device's hardware.

In conclusion, it is important to consider the computer graphic rendering results but also have platform efficiency in mind. Diverdi and team provided us with an efficient algorithm and app to create a watercolor painting app that was able to run on the IPad. It is paramount to provide access to commuter graphics apps to many different platforms. That is because mobile devices are growing faster than stationary devices so it is important to make our creation available to all users regardless of platform and device type.

Bibtex

Article title: Democratizing Digital Content Creation Using Mobile Devices with Inbuilt Sensors

@ARTICLE{7030191,
author={Dev, K. and Lau, M.},
journal={Computer Graphics and Applications, IEEE},
title={Democratizing Digital Content Creation Using Mobile Devices with Inbuilt Sensors},
year={2015},
month={Jan},
volume={35},
number={1},
pages={84-94},
url={http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=7030191}
keywords={content management;mobile computing;content production activity;digital content creation;inbuilt sensors;mobile content creation;mobile devices;ubiquitous computing;Accelerometers;Digital systems;Mobile communication;Mobile handsets;Painting;Sensors;Three-dimensional displays;computer graphics;digital content creation;mobile graphics;mobile interaction;mobile sensors;mobile user interface},
doi={10.1109/MCG.2015.10},
ISSN={0272-1716},}

Article title: Painting with Polygons: A Procedural Watercolor Engine

@ARTICLE{6314479,
author={DiVerdi, S. and Krishnaswamy, A. and Mech, R. and Ito, D.},
journal={Visualization and Computer Graphics, IEEE Transactions on},
title={Painting with Polygons: A Procedural Watercolor Engine},
year={2013},
month={May},
volume={19},
number={5},
pages={723-735},
url={http://ieeexplore.ieee.org.libproxy.uml.edu/stamp/stamp.jsp?tp=&arnumber=6314479},
keywords={art;geometry;image resolution;painting;pigments;rendering (computer graphics);screens (display);vectors;arbitrary resolutions;digital artists;dynamic behaviors;natural media painting simulation;polygons;print resolution;procedural pigment advection algorithm;procedural watercolor engine;screen resolutions;slate devices;vector based stroke representation;vector-rendering technologies;watercolor-like dynamic paint behaviors;watercolor-like painting tools;Brushes;Heuristic algorithms;Painting;Paints;Pigments;Tablet computers;Vectors;Natural media;real time;vector graphics;watercolor painting;Algorithms;Color;Computer Systems;Imaging, Three-Dimensional;Paint;Paintings;Software;User-Computer Interface},
doi={10.1109/TVCG.2012.295},
ISSN={1077-2626},}