IARIA

CONTENT 2016

The Eighth International Conference on Creative Content Technologies

ISBN: 978-1-61208-464-0

March 20 - 24, 2016

Rome, Italy

CONTENT 2016 Editors

Hans-Werner Sehring, Namics AG, Germany

René Berndt, Fraunhofer Austria Research GmbH, Austria

CONTENT 2016

Forward

The Eighth International Conference on Creative Content Technologies (CONTENT 2016), held between March 20-24, 2016 in Rome, Italy, continued a series of events targeting advanced concepts, solutions and applications in producing, transmitting and managing various forms of content and their combination. Multi-cast and uni-cast content distribution, content localization, on-demand or following customer profiles are common challenges for content producers and distributors. Special processing challenges occur when dealing with social, graphic content, animation, speech, voice, image, audio, data, or image contents. Advanced producing and managing mechanisms and methodologies are now embedded in current and soon-to-be solutions.

The conference had the following tracks:

- Image and graphics
- Web content
- Content producers/distributors

Similar to the previous edition, this event attracted excellent contributions and active participation from all over the world. We were very pleased to receive top quality contributions.

We take here the opportunity to warmly thank all the members of the CONTENT 2016 technical program committee, as well as the numerous reviewers. The creation of such a high quality conference program would not have been possible without their involvement. We also kindly thank all the authors that dedicated much of their time and effort to contribute to CONTENT 2016. We truly believe that, thanks to all these efforts, the final conference program consisted of top quality contributions.

Also, this event could not have been a reality without the support of many individuals, organizations and sponsors. We also gratefully thank the members of the CONTENT 2016 organizing committee for their help in handling the logistics and for their work that made this professional meeting a success.

We hope CONTENT 2016 was a successful international forum for the exchange of ideas and results between academia and industry and to promote further progress in the area of creative content technologies. We also hope that Rome provided a pleasant environment during the conference and everyone saved some time for exploring this beautiful city.

CONTENT 2016 Chairs

CONTENT Advisory Chairs

Raouf Hamzaoui, De Montfort University - Leicester, UK
Jalel Ben-Othman, Université de Versailles, France
Jaime Lloret Mauri, Polytechnic University of Valencia, Spain
Wolfgang Fohl, Hamburg University of Applied Sciences, Germany
Zhou Su, Waseda University, Japan

CONTENT Industry/Research Chairs

Ajith Abraham, Machine Intelligence Research Labs (MIR Labs), USA
Hans-Werner Sehring, Namics AG, Germany
René Berndt, Fraunhofer Austria Research GmbH, Austria

CONTENT Publicity Chairs

Lorena Parra, Universidad Politécnica de Valencia, Spain
Samuel Kosolapov, Braude Academic College of Engineering, Israel
Wilawan Inchamnan, Queensland University of Technology, Australia
Javier Quevedo-Fernandez, Eindhoven University of Technology, The Netherlands

CONTENT 2016

Committee

CONTENT Advisory Committee

Raouf Hamzaoui, De Montfort University - Leicester, UK
Jalel Ben-Othman, Université de Versailles, France
Jaime Lloret Mauri, Polytechnic University of Valencia, Spain
Wolfgang Fohl, Hamburg University of Applied Sciences, Germany
Zhou Su, Waseda University, Japan

CONTENT Industry/Research Chairs

Ajith Abraham, Machine Intelligence Research Labs (MIR Labs), USA
Hans-Werner Sehring, Namics AG, Germany
René Berndt, Fraunhofer Austria Research GmbH, Austria

CONTENT Publicity Chairs

Lorena Parra, Universidad Politécnica de Valencia, Spain
Samuel Kosolapov, Braude Academic College of Engineering, Israel
Wilawan Inchamnan, Queensland University of Technology, Australia
Javier Quevedo-Fernandez, Eindhoven University of Technology, The Netherlands

CONTENT 2016 Technical Program Committee

Naveed Ahmed, University of Sharjah, UAE
Jose Alfredo F. Costa, Federal University (UFRN), Brazil
Marios C. Angelides, Brunel University - Uxbridge, UK
Kambiz Badie, Research Institute for ICT & University of Tehran, Iran
David Banks, University of Tennessee, USA
Chedi Bechikh, Université de Tunis - Institut supérieur de Gestion, Tunisia
Rene Berndt, Fraunhofer Austria Research GmbH, Austria
Christos Bouras, University of Patras and Computer Technology Institute & Press «Diophantus», Greece
Oliver Bown, University of Sydney, Australia
Andrew Brown, Queensland University of Technology, Australia
Yiwei Cao, IMC AG, Germany
Wojciech Cellary, Poznan University of Economics, Poland
Lijun Chang, University of New South Wales, Australia
Savvas A. Chatzichristofis, Democritus University of Thrace, Greece
Chi-Hua Chen, National Chiao Tung University, Taiwan, R.O.C.
Octavian Ciobanu, "Gr.T. Popa" University of Medicine and Pharmacy – Iasi, Romania
Raffaele De Amicis, Fondazione Graphitech - Trento, Italy

Marc Tomlinson, Language Computer Corporation, USA
Paulo Urbano, University of Lisbon, Portugal
Anna Ursyn, University of Northern Colorado, USA
Stefanos Vrochidis, Information Technologies Institute, Greece
Krzysztof Walczak, Poznan University of Economics, Poland
Stefan Wesarg, Fraunhofer IGD - Darmstadt, Germany
Wojciech R. Wiza, Poznan University of Economics, Poland
Shigang Yue, University of Lincoln, UK
Juan Zamora, Universidad Técnica Federico Santa María, Chile

Copyright Information

Table of Contents

Self-Driven Soft-Body Creatures

Ben Kenwright
School of Computing
Edinburgh Napier University, UK
Email: b.kenwright@napier.ac.uk

Kanida Sinmai
Department of Computer and Information Technology
Thaksin University, Thailand
Email: kanida@tsu.ac.th

***Abstract*—Virtual characters play an important role in computer-generated environments, such as, video games, training simulations, and animated films. Traditional character animation control methods evolve around key-frame systems and rigid skeletons. In this paper, we investigate the creation and control of soft-body creatures. We develop creatures that learn their own motor controls and mimic animal behaviours to produce autonomous and coordinated actions. Building upon passive physics-based methods and data-driven approaches, we identify solutions for controlling selective mesh components in a coherent manner to achieve self-driven animations that possess plausible life-like characteristics. *Active soft-body animations* open the door to a whole new area of research and possibilities, such as, morphable topologies, with the ability to adapt and overcome a variety of problems and situations to accomplish specified goals. We focus on two and three-dimensional deformable creatures that use physics-based principles to achieve *unconstrained self-driven motion* as in the real-world. As we discuss, control principles from passive soft-body systems, such as, clothes and finite element methods, form the foundation for more esoteric solutions. This includes, controlling shape changes and locomotion, as movement is generated by internally changing forces causing deformations and motion. We also address computational limitations, since theoretical solutions using heuristic models that train learning algorithms can have issues generating plausible motions, not to mention long search times for even the simplest models due to the massively complex search spaces.**

Keywords*–animation, control, soft-bodies, characters, motion, physics, deformation, creatures, movement, unconstrained, physics-based, self-driven

I. Introduction

Movement without a Skeleton Soft-body creatures are organisms or animals that lack a skeleton (a real-world soft-body example, would be a leech, a jellyfish, or a even a tongue). The question of how to efficiently represent the creature and create 'controlled' movement is an open topic of research. As soft-body creatures do not have the luxury of an internal **musculo-skeleton** system to guide and steer their motion, but must instead generate movement solely on the contraction and expansion of their body tissues. Another key thing to remember, is soft-body creatures are not confined to anatomically-based structures (such as, bipeds or quadrupeds) and provide a means of freedom and creativity. The creation and controlling of self-driven soft-body creatures with scalable properties that learn their own motor controls and mimic animal behaviors to produce autonomous and coordinated actions is an important and challenging subject.

Control & Realism Skeleton key-frame methods are the dominant solution for creating controlled creature animations. As they offer an intuitive, flexible and powerful solution that can produce highly realistic results. However, can skeleton based models generate animations that move and look the same as a soft-body system? Assuming a soft-body creature must keep its overall volume, a soft-body system is able to squeeze and reduces its circumference and stretch to increases its length. The internal deformations generate movement and provide a visible set of secondary visual characteristics that add a natural aura to the motion that are not apparent in purely rigid simulation solutions.

Contribution This paper presents a soft-body creature control system with scalable properties (i.e., trade-offs between computational speed and detail). Our approach makes the following technical contributions: (1) generation of steerable deformations that control a character's soft-body motions to achieve targeted animations under their own forces (unconstrained movements through internal forces and contacts with the environment); (2) the model does not require any intensive off-line pre-processing, enabling artists to re-iterate multiple versions quickly and efficiently to develop more creative and imaginative solutions (enabling artistic freedom); and (3) we create controlled animations that interact with the physical simulation (e.g., push disturbances and secondary motions, such as, vibrations and ripples).

Road Map The rest of the paper is structured as follows: First, Section II discusses related work. In Section III, we give a explanation of our algorithm and practical considerations for real-time environments. Section IV presents the results from the simulation examples. Section V discusses the implication of the method and explains any problems after taking the results into consideration. Finally, Section VI draws conclusions from the approach and future work.

II. Related Work

There has been lots of work into 'passive' soft-body systems, such as, skin and cloth [1], [2]. These techniques provide a wealth of information that we build upon in this paper to construct animated soft-body creatures. The recent work by Cheney et al. [3] and Kenwright [4] inspired the research behind in this paper. Presenting innovative solutions that go beyond traditional key-frame based methods towards more esoteric procedural ones. For example, this includes, a muscle-driven solution by Tan et al. [5] who creates animated soft-bodies using an action line and helical muscle fibers for twisting movements. Combined with key-frame data prescribed by an animator to steer and direct the resulting simulations and generate life-like deformations (i.e., primary motions for movement and secondary motions for appeal).

The area of soft-body creatures covers a diverse range of topics across multiple disciplines. For instance, soft-body creatures does not mean just 'land' creatures, but can also be controlled muscle segments, like the tongue, not to mention flying and swimming animals. A good example of this, is the work by Tan et al. [6] who did ground breaking work into soft-body fish simulations. The research in soft-body creatures also

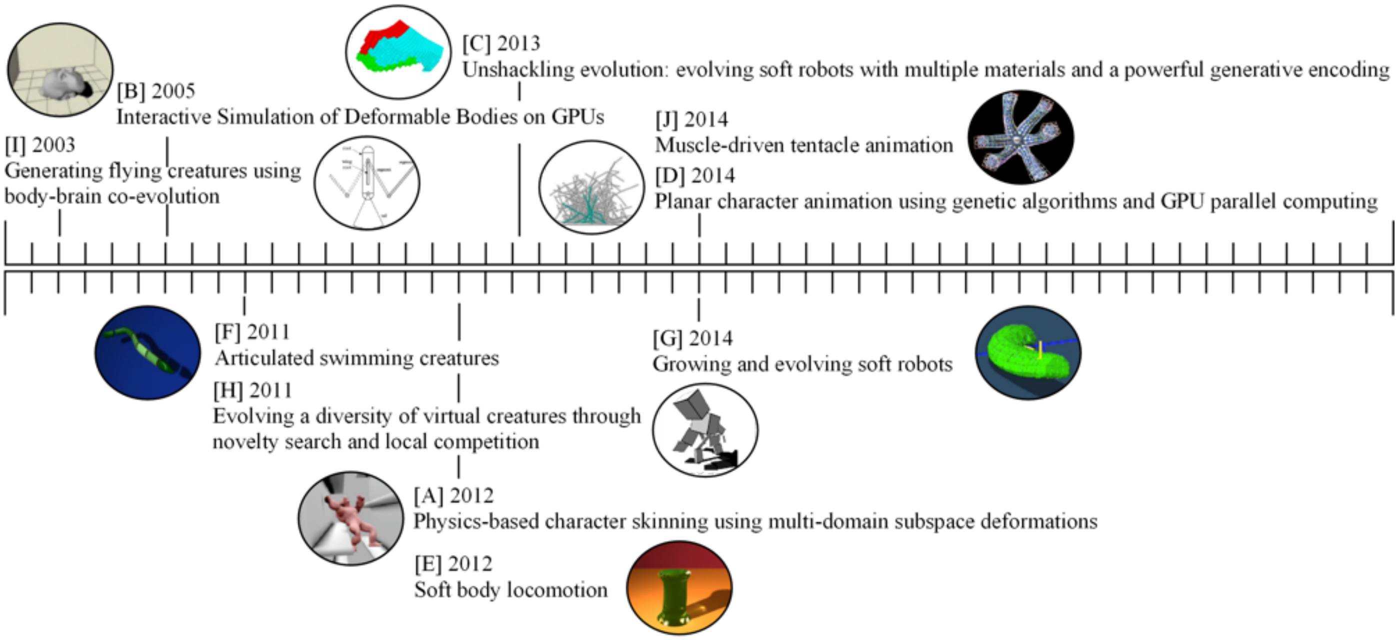

[C] 2013
Unshackling evolution: evolving soft robots with multiple materials and a powerful generative encoding
[B] 2005
Interactive Simulation of Deformable Bodies on GPUs
[J] 2014
Muscle-driven tentacle animation
[I] 2003
Generating flying creatures using
body-brain co-evolution
[D] 2014
Planar character animation using genetic algorithms and GPU parallel computing
[F] 2011
Articulated swimming creatures
[G] 2014
Growing and evolving soft robots
[H] 2011
Evolving a diversity of virtual creatures through
novelty search and local competition
[A] 2012
Physics-based character skinning using multi-domain subspace deformations
[E] 2012
Soft body locomotion

Figure 1. **Timeline** - Visual illustration of related publications over the past few years that have contributed towards more active and engaging soft-body systems. [A] [1], [B] [2], [C] [3], [D] [4], [E] [5], [F] [6], [G] [7], [H] [8], [I] [9], [J] [10]

spins out to real-world models, such as, robots that grow and evolve [7]. The paper 'Unshackling Evolution: Evolving Soft Robots with Multiple Materials and a Powerful Generative Encoding' by Cheney et al. [3] used a technique that divided the object into 'components' (i.e., voxels) to create the controlled motion. Then there was Lehman and Stanley [8], who evolved a diverse range of creatures through a novelty search and local competition. Shim and Kim [9] created flying creatures using body brain co-evolution, while Liu et al. [11] focused on a hybrid skeleton to control the soft-bodies. A number of these soft-body techniques have employed the 'action-line' control system in combination with an optimisation search to steer and direct the animation [5], [12]. For instance, a muscle-driven tentacle animation by Stavness [10] and the soft-body locomotion by Tan et al. [5]. In conclusion, active self-driven soft-body creatures offer a novel solution for solving a wide range of potential problems (see Figure 1 for a brief visual illustration of inspiring research on the topic over the past few years).

Our work: Our work focuses on the ability to create imaginary and non-imaginary topologies and have them move in a controlled manner (e.g., hop or walk along) driven under their own internally changing forces. We incorporate techniques from passive physics-based simulation systems and coarse control meshes to reduce computational costs. While we use an underpinning physics-based model, we integrate in control approximations, such as, coordinated rhythmic oscillations, to control and direct the soft-body's movement towards an organic and aesthetically pleasing solution.

III.METHOD

A purely procedural soft-body solution is plausible using heuristic algorithms. For example, training trigonometric functions (e.g., any signal can be composed of sinusoidal signals - the concept behind Fourier series [13]). However, this opens the door to a vast array of parameters with a finite search range that is difficult to solve in viable time frames and allow for artistic control. Alternatively, we exploit a smarter solution uses a hybrid combination of methods, such as, pre-recorded training data to steer the system towards approximate solutions, in combination with human intervention to create self-driven soft-body actions. We take animation data (i.e., pre-recorded key-frames). We connect the soft-body to the skeleton via distance constraints, so as the animation plays the points will move in a correlated pattern. As an important factor is the creation of soft-body motions that are controllable (rather than just randomly jiggling around). The kinematic solution of the coupled system provides a starting set of oscillating distance constraints from which we can calculate the penalty forces and inverse dynamics. When we play back the soft-body system using the calculated forces. The motions will be driven by the physical system, and will only be approximate, but will capture a starting essence for the animation (based on the key-frame data). Of course, the motion will drift away from the predefined target set out by the key-frame data (the animation will fall over or wobble to one side). This is where we need to adapt and adjust the forces to steer and control the final animation. Allowing an animator to use key-frame motion initially to target and formulate the underlining motion for the soft-body and aid in the rapid proto-typing and development (see Figure 2). Since the final simulation is generated using physics-based concepts, the deformations are organic with directable purpose (i.e., walk or move in a desired way - artistic influence).

Our physically-based approach to soft-body animation evolves around the automatic control of the contractions/expansions of interconnected constraints. The internal forces drive the soft-body model's motion. The formulation

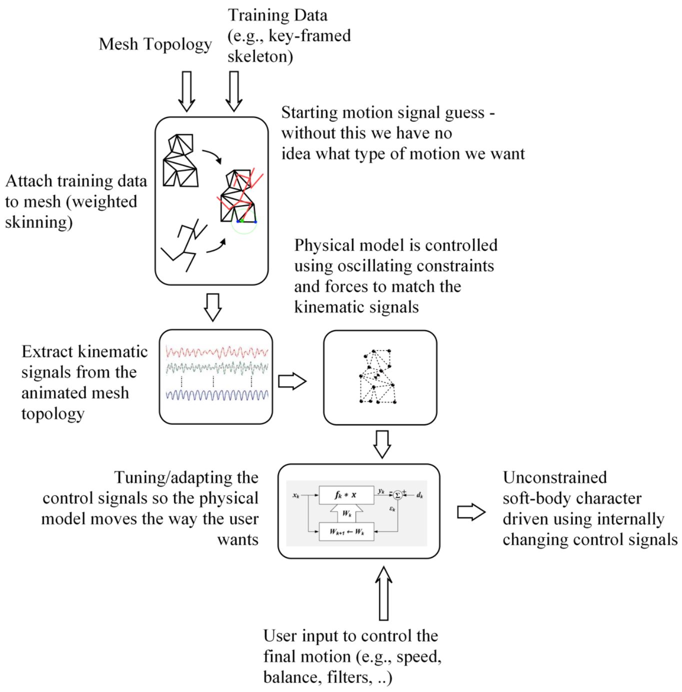

Mesh Topology
Training Data (e.g., key-framed skeleton)
Starting motion signal guess - without this we have no idea what type of motion we want
Attach training data to mesh (weighted skinning)
Physical model is controlled using oscillating constraints and forces to match the kinematic signals
Extract kinematic signals from the animated mesh topology
Tuning/adapting the control signals so the physical model moves the way the user wants
Unconstrained soft-body character driven using internally changing control signals
User input to control the final motion (e.g., speed, balance, filters, ..)

Figure 2. **Overview** - Interconnected elements to construct and tune our soft-body character's unconstrained motions.

of a passive soft-body system is straightforward, with the challenges evolving around the control of the internal forces to achieve directed motion. Penalty-based methods and inverse dynamic calculations allow us to solve for desired forces to accomplish positional changes. **Combining animator controlled targets** with a dynamic force driven simulation, we capture the global movement specifications of the animator (artistic characteristics) and achieve detailed body deformation effects that appears organic and life-like (i.e., analogues to a real-world organic creature). We emulate a secondary underlying deformation motions that include squeezing and bulging through lengthening, bending, and twisting movements in a free moving action **without an internal skeleton**. We use a point-mass structure with a spring-damper system for experiments that do not require exact solutions (i.e., quick proto-typing/investigation). For more precise solutions (i.e., conservation of momentum and stiff-constraints), we use a finite element composition and a constraint based solver. Decomposing the mesh into a finite-element structure (triangles in 2D and tetrahedrons in 3D). Oscillating the rest forces between the constraints to create rhythmic movements.

A. *Motion Mechanics*

We explore and propose two methods for generating soft-body motions, i.e., procedural (trigonometric) and data-driven (signal-data).

Trigonometric (Fourier Series) For some systems, a goal is specified, such as, walking speed or jumping height. For a highly coupled system, it can be difficult to achieve a unified motion to accomplish such an action. Applying the concept that any signal can be represented by a series of sinusoidal signals. We use an optimisation algorithm to train the trigonometric parameters (i.e., amplitude, frequency and offset) to achieve a motion which accomplishes a specified fitness criteria. The output from the trigonometric solution may also form a good approximation for artists to start from. The signals from the trigonometric functions can be saved and modified to create more aesthetic solutions.

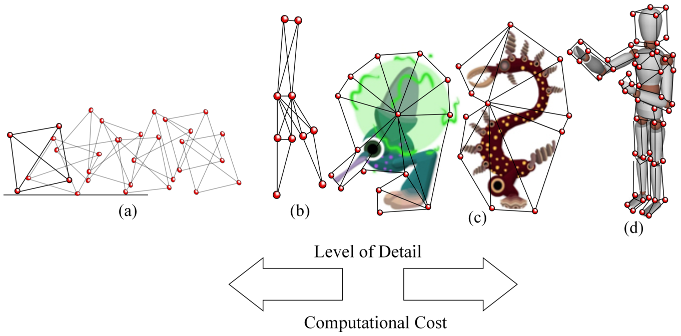


Figure 3. **Animating Soft-Body System** - Decomposing the model into elements (e.g., point-masses) and controlling the interconnected elements to achieve directed animations.

Trained/Customized Signals The ability to allow an artist to customize and adapt constraint signals is important for freedom of creativity. For example, it is difficult to train soft-body motion signals using purely key-frame data, as animations, such as, jumping - require the body to squash and conserve energy then release it to achieve a jumping motions. Additionally, if a signal is created by an artist, we are able to use Fourier transform to extract parameters or even filter components to target different motions.

B. *Overall Volume & Shape*

A point-mass system offers a simple and intuitive solution for the majority of cases. However, there are situations that require the shape to be distorted, while preserving the overall volume. For these situations, a finite element method (FEM) is required (i.e., area and Young's modulus calculation). FEM keeps track of the internal volume by distribution of pressure forces outwards via the connecting corner points (i.e., assuming a triangle or tetrahedron configuration).

For a point-mass system with distance constraints, we limit contraction/expansion, (e.g., 10% to keep the overall appearance and some form of the original shape) in addition to limiting forces, while for volume regulation using an FEM topology, each partitioned segment must retain its original volume, within some tolerance, allowing the shape to deform and stretch (for instance, taking a small fat object and stretching it into a thin long one, analogues to a leech contracting and stretching but continuing to hold the same volume).

C. *Low-Resolution Control Mesh*

We explain our low-dimensional control model for creating controlled deformations and ultimately the goal of self-driven soft-body creature animations. A coarse control mesh technique is used to reduce the computational overhead and the mathematical complexity of the model, so we are able to achieve real-time frame rates and quick turn-around times. We explain how the high-resolution graphical mesh interactions with the coarser control mesh and how contacts between the mesh and the virtual environment are handles (i.e., in an endeavour to realistically mimic the mechanical deformation properties of organic tissue).

Mesh embedding, which is also called *free-form* deformation [14], [15], uses a low-dimensional coarse volumetric mesh to enclose the entire deformable body in order to represent the behavior of the body. The location of every material point inside the deformable body is determined by interpolating the positions of the neighboring nodes in the mesh. Since the work by Faloutsos et al. [16], mesh embedding techniques have been widely used to simulate soft-bodies in the graphics literature [17]–[19]. We chose mesh embedding to reduce complexity of the deformable body in our simulation system not only because the technique can reduce the model complexity without losing the fine geometry of the object but also because the frame can be manipulated more easily and efficiently using the embedding mesh system compared to modal reduction. In our formulation, the control body is the core soft-body system that drives the deformations and ultimately the animation. The complete system consists of a set of deformable body elements (i.e., tetrahedrons or voxels) and a physics-based soft-body core (see Figure 4).

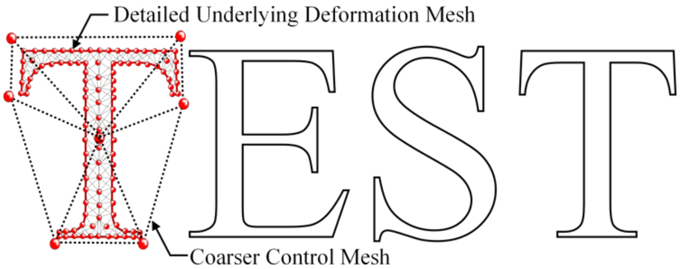


Figure 4. **Coarse Control Mesh** - Reduce the complexity of the problem while preserving the underlying soft-body's freedom.

The position of a material point in the deformable body is determined from the nodal positions of the coarse mesh through interpolation. Initially, we automatically assigned weights and blended the vertices based on the inverse distance between the coarse mesh control nodes and fine mesh vertices (i.e., distance-based falloff weighting). However, since the mesh is partitioned into regions (cells), triangles in 2D and tetrahedrons in 3D, it is possible to assign vertices into region. For example, every point inside a tetrahedron can be expressed as a linear combination of the four control vertices. For every vertex p in the high-resolution mesh, we look up the tetrahedron that the point is in, which has four control vertices (call them v_0, v_1, v_2, v_3). If we translate every vector by $-v0$, we can write the new point $p' = p - v_0$ as a linear combination of v'_1, v'_2, and v'_3. This gives us a new equation for p', namely $p' = u\ v'_1 + v\ v'_2 + w\ v'_3$, where u, v and w are the weights (i.e., Barycentric coordinates). Now we can write the point p as $v_0 + u(v_1 - v_0) + v(v_2 - v_0) + w(v_3 - v_0)$. Deforming the tetrahedron mesh, v_0, v_1, v_2 and v_3 will change p accordingly. This deformation will depend on the four nearby vertices, and thus a deformation on one side of the mesh will have no effect on the other side of the high-resolution mesh (unlike the distance-based falloff, where every vertex affects every other vertex in the high-resolution mesh - see Kenwright [14] for further information).

Level of Detail (LOD) Adjusting the complexity of the problem by subdividing the mesh into a coarser 'control' submeshes allows us to scale the complexity of the problem. This analogy is similar to the work by Kim and Pollard [15], who focused on fast and efficient skeleton-driven deformable body characters, however, our model uses the underlying deformation to achieve the controlled creature motions (Figure 3).

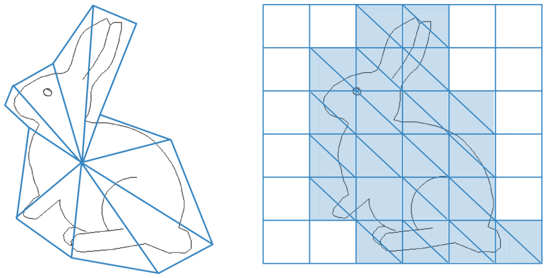


Figure 5. **Elements** - Variety of coarse mesh configurations, the two main ones we focused on, were tetrahedrons (left) and voxels (right) (e.g., Figure 8 shows the voxelated gummy-bear mesh).

Scalability We endeavour to automate the modelling/creation process as much as possible, and reduce the amount of artist workload by providing a system that enables both productivity and creativity. A low-poly control mesh enables artists to easily proto-type key-frame motions quickly and easily in real-time (see Figure 4 and Figure 5). Proving an artistic tool for investigating deformation effects and adjusting them to different platforms (i.e., reducing the model complexity to more coarser representations for environments with limited resources, such as, memory and processing power).

IV. Experimental Results

Experimenting with a low-dimensional 2D model initially provides the basis for more complex systems. Adjusting the mesh/point-mass/constraint details in 2D and 3D, combined with oscillatory rhythmic input from trigonometric sources and pre-trained signals from either an artist or from motion capture data. This includes, simple hopping and wiggling locomotion for cubes to more complex structured motions with a larger interconnected set of constraints. Simulations:

- simple 2-dimensional square bouncing along ground (i.e., forward and backward) - the motion is generated by contracting and expanding the edges of the square in a controlled unified manner (see Figure 6)
- more complex 2-dimensional shapes (e.g., convex and concave)
- 3-dimensional shapes (i.e., cubes and tetrahedrons) (see Figure 7)
- coarser models (i.e., lower dimensional control shapes creating the motion combined with a higher resolution mesh) (see Figure 8 and Figure 9)

Preliminary experiments evolved around 'rhythmic' algorithms using trigonometric functions (i.e., Fourier series). Truncating series of sinusoidal signals with different parameters for the different constraints, makes it is easy to construct simple gait motions (i.e., low-dimensional models like cubes and pyramids). Crucial factors are 'friction' and 'constraint fighting'. Not only is friction important to prevent the model simply skate on the spot, but also the ‘unified’ cooperation of the different signals. Manually adjusting parameters to achieve controlled action are possible for simple models, but difficult for more complex systems. The simulations demonstrate the character’s ability to adapt and learn their own motor controls to mimic animal behaviours and create autonomous and coordinated actions. Finally, since we are training a coarser low-dimensional control mesh, it helps address computational bottlenecks.

Limitations A system of interconnected nodes representing the soft-body mechanism can grow in complexity quickly and make it difficult to formulate and train a controlled solution to accomplish directed actions. An interesting area of future exploration is that of massively parallel architecture to divide the workload of the problem across a large number of smaller cores (e.g., the graphical processing unit (GPU)). Coupled with the question of ‘comfort’ and ‘shape’, since we may want our creatures to move in an organic and relaxed manner while keeping their overall form. Of course, this may not be the case - since the approach allows the ability to change shape, possibly to overcome environmental situations or accomplish some artistic desire, which is important and challenging.

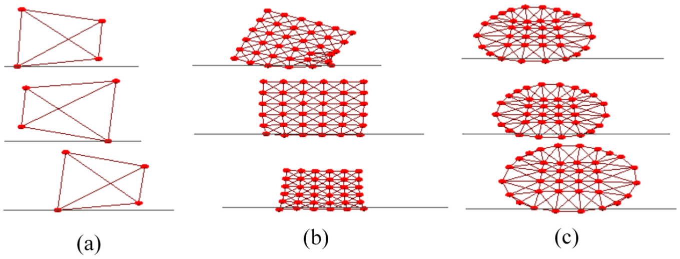


Figure 6. **Screenshot** - (a) A 4-point mass-spring topology using coordinated oscillating constraints to create a rhythmic gait like motion as seen by quadrupeds. (b) Increasing the mass distribution enables more complex deformations and motions but introduces additional difficulties creating ‘coordinated’ movement (i.e., avoiding constraint fighting). (c) Experimenting with shapes, like circles, we can create stepping motions analogous to the box-quadruped system, or rolling like motion as seen by a vehicle tyre.

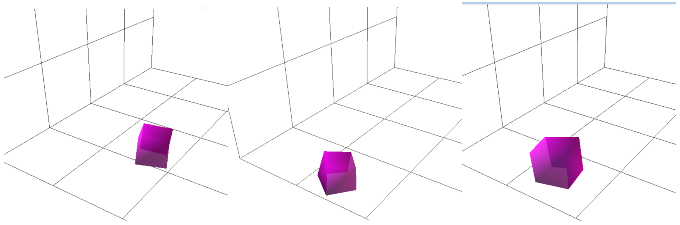


Figure 7. **Screenshot** - For low-poly shapes with few-constraints, the oscillations for gait like motion can be modified on the fly to create steered motion for navigating a virtual world.

V. Discussion

While this paper has focused on a soft-body system, combining our approach with an underlying rigid-body skeleton (hybrid technique) would open the door to further advantages (i.e., the explicit injection of key-frame motions). For example, a motion capture method would control a rigid-body skeleton from which joint torques are calculated to mimic the motion. However, the joint torques are adjusted based on the environment and model’s state (e.g., balance). Adding an active soft-body system on top of the rigid skeleton - changing the overall centre of mass and injecting rhythmic motions into the movement, such as, breathing and swaying, to produce an animation that is closer to what we see in real-world organic creatures. While the model is decomposed of ‘elements’, it also opens the door to subdivision and fractured motion (e.g., an object that is able to split into smaller soft-body elements, each possessing their own individual motion mechanics). The subdivided mesh remains fixed, however, an area of further investigation would be the dynamic adaptation of the resolution as the character mesh deforms adding and removing extra detail where needed.

In summary, the technique in this paper presents a number of implementation difficulties in practical applications. For example, without sufficient constraints (i.e., addition structural, sheer, and bending constraints), would result in the mesh being unable to support itself and the deformations would be rough and coarse with abrupt and sharp edges. However, our technique can be combined with different model reduction control methodologies, to exploit the computational power of the GPU and help reduce the complexity and computational limitations for more detailed tasks/geometry.

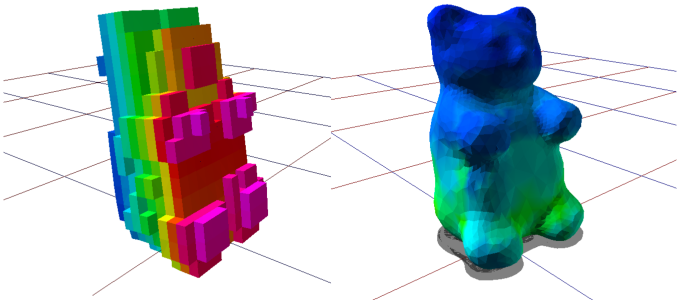


Figure 8. **Gummy-Bear Coarse Voxel Mesh** - A coarse over-mesh decomposed of voxels allows provides a computational speed-up.

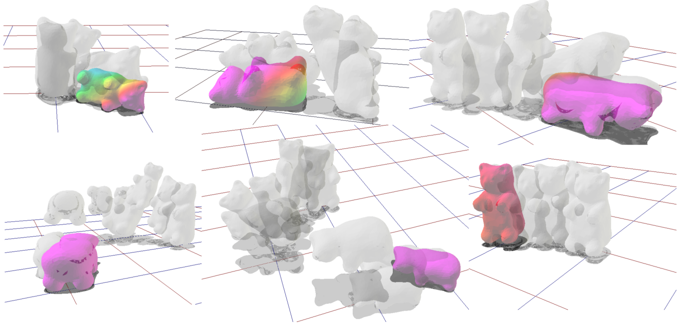


Figure 9. **Gummy-Bear** - Training and deforming the constrains of a gummy-bear mesh to achieve targeted motions, such as, balanced hopping or forward crawling motions. Through experimentation, we are able to create a diverse assortment of solutions, e.g., simple wiggling, bouncing, and shuffling.

VI. Conclusion

Internal forces cause deformations and ultimately structural changes that animate the model. External contacts with the

Copyright (c) IARIA, 2016. ISBN: 978-1-61208-464-0

environment provide reactive collision and contact forces that enable the soft-body system to explore its virtual world. We have reviewed and explored a number of implementation factors, such as, multi-resolution meshes, volume conservation, and animation control, using search driven fitness optimization algorithms in combination with artistic intervention and key-frame data. The model forms the basis of other hybrid systems that mix rigid and soft-body systems, such as, muscles and skin.

ACKNOWLEDGEMENTS

We would like to thank the reviewers for taking time out of their busy schedules to provide constructive and insightful feedback that has went into making this paper more clear and concise.

REFERENCES

[1] T. Kim and D. L. James, “Physics-based character skinning using multidomain subspace deformations,” Visualization and Computer Graphics, IEEE Transactions on, vol. 18, no. 8, 2012, pp. 1228–1240.

[2] J. Georgii, F. Echtler, and R. Westermann, “Interactive simulation of deformable bodies on gpus.” in SimVis, 2005, pp. 247–258.

[3] N. Cheney, R. MacCurdy, J. Clune, and H. Lipson, “Unshackling evolution: evolving soft robots with multiple materials and a powerful generative encoding,” in Proceedings of the 15th annual conference on Genetic and evolutionary computation. ACM, 2013, pp. 167–174.

[4] B. Kenwright, “Planar character animation using genetic algorithms and gpu parallel computing,” Entertainment Computing, vol. 5, no. 4, 2014, pp. 285–294.

[5] J. Tan, G. Turk, and C. K. Liu, “Soft body locomotion,” ACM Transactions on Graphics (TOG), vol. 31, no. 4, 2012, p. 26.

[6] J. Tan, Y. Gu, G. Turk, and C. K. Liu, “Articulated swimming creatures,” in ACM Transactions on Graphics (TOG), vol. 30, no. 4. ACM, 2011, p. 58.

[7] J. Rieffel, D. Knox, S. Smith, and B. Trimmer, “Growing and evolving soft robots,” Artificial life, vol. 20, no. 1, 2014, pp. 143–162.

[8] J. Lehman and K. O. Stanley, “Evolving a diversity of virtual creatures through novelty search and local competition,” in Proceedings of the 13th annual conference on Genetic and evolutionary computation. ACM, 2011, pp. 211–218.

[9] Y.-S. Shim and C.-H. Kim, “Generating flying creatures using body-brain co-evolution,” in Proceedings of the 2003 ACM SIGGRAPH/Eurographics symposium on Computer animation. Eurographics Association, 2003, pp. 276–285.

[10] I. Stavness, “Muscle-driven tentacle animation,” in SIGGRAPH Asia 2014 Posters. ACM, 2014, p. 36.

[11] L. Liu, K. Yin, B. Wang, and B. Guo, “Simulation and control of skeleton-driven soft body characters,” ACM Transactions on Graphics (TOG), vol. 32, no. 6, 2013, p. 215.

[12] C. Schulz, C. von Tycowicz, H.-P. Seidel, and K. Hildebrandt, “Animating deformable objects using sparse spacetime constraints,” ACM Transactions on Graphics (TOG), vol. 33, no. 4, 2014, p. 109.

[13] B. Kenwright, “Quaternion fourier transform for character motions,” in Workshop on Virtual Reality Interaction and Physical Simulation, VRIPHYS 2015, Lyon, France, November 4-5, 2015., 2015, pp. 1–4.

[14] ——, “Free-form tetrahedron deformation,” in Advances in Visual Computing - 11th International Symposium, ISVC 2015, Las Vegas, NV, USA, December 14-16, 2015, Proceedings, Part II, 2015, pp. 787–796. [Online]. Available:

[15] J. Kim and N. S. Pollard, “Fast simulation of skeleton-driven deformable body characters,” ACM Transactions on Graphics (TOG), vol. 30, no. 5, 2011, p. 121.

[16] P. Faloutsos, M. Van De Panne, and D. Terzopoulos, “Dynamic free-form deformations for animation synthesis,” Visualization and Computer Graphics, IEEE Transactions on, vol. 3, no. 3, 1997, pp. 201–214.

[17] M. Nesme, P. G. Kry, L. Jeřábková, and F. Faure, “Preserving topology and elasticity for embedded deformable models,” in ACM Transactions on Graphics (TOG), vol. 28, no. 3. ACM, 2009, p. 52.

[18] L. Kharevych, P. Mullen, H. Owhadi, and M. Desbrun, “Numerical coarsening of inhomogeneous elastic materials,” in ACM Transactions on Graphics (TOG), vol. 28, no. 3. ACM, 2009, p. 51.

[19] S. Capell, M. Burkhart, B. Curless, T. Duchamp, and Z. Popović, “Physically based rigging for deformable characters,” in Proceedings of the 2005 ACM SIGGRAPH/Eurographics symposium on Computer animation. ACM, 2005, pp. 301–310.

Subjective Assessment of Face Photographs by Best-Worst Method

How Contracdictive Factors are Evaluated When We See Images

Kyosuke Shimada, Hirohisa Takeshita and Seiichi Gohshi
Department of Informatics
Kogakuin University
Tokyo, Japan
E-mail: em15009@ns.kogakuin.ac.jp, takeshita1312@gmail.com, gohshi@cc.kogakuin.ac.jp

***Abstract*— Human faces are commonly photographed. We know from experience that we are attracted to faces when we appreciate personal photographs. Therefore, it is clear that personal photographs have a different type of interest from landscape photographs in subjective assessment. Finding interesting aspects of such photographs can improve the conditions of taking photographs. These conditions provide a significant guideline for developing digital cameras that capture high-quality photographs. Investigation of interesting factors requires image assessment considering noise and resolution because those are the most important factors of image quality determining in the assessment. This study reports important factors of image quality assessment based on the investigation of processes assessed by resolution and noise on human face photographs.**

Keywords- Subjective assessment; Human face; Noise; Resolution.

I. INTRODUCTION

A. Background

Imaging technology has enabled video equipment including digital cameras to capture high-resolution images in inferior lighting conditions and printed photographs of such images have been high quality. Luminance of images is determined by the sensitivity of the complementary metal-oxide semiconductor (CMOS) image sensor [1] and lighting conditions. Resolution depends on the number of image sensors. The International Standards Organization (ISO) [2] number is an international standard for sensitivity of image sensors. Recent digital cameras can select a wide range of sensitivities from ISO 100 to ISO 6400 for taking images. The image quality is changed by setting the ISO sensitivity and the F-number in the same lighting condition. The F-number affects the luminance of an image. The image quality differences due to setting fluctuations appear on images as resolution and noise. It is difficult to evaluate images taken with changing F-number and ISO sensitivity because the difference in the photographic quality is too small.

The conditions for high quality are high-resolution and low-noise. Various noise reduction methods have been used to reduce noise. However, noise and resolution are contradictory factors because resolution decreases with reduction in noise.

Noise is an essential issue for images because it degrades image quality. Even though sensitivity of imaging devices has dramatically improved in recent years, all images taken with imaging devices contain certain levels of noise. Resolution, which is a contradicting factor with relation to noise, is also an important element when assessing the image quality. Noise and resolution are composed with high frequency elements and it is impossible to divide them with digital signal processing. Due to this reason, we have to assess image quality in accordance with the balance of noise and resolution.

There are two methods to assess image quality. They are objective assessments and subjective assessments. Although the objective assessment is reproducible and reliable, it is difficult to apply it to images with noise and high resolution. In contrast the subjective assessment is a practical method to assess image quality. However, the subjective assessment has not yet been applied to assess the image quality that has contradictory factors, noise and resolution. The subjective assessment comes with advantage of reproducibility. It can be guaranteed with statistical analysis. We have proposed the Best-Worst Method (BWM) to assess images that have high resolution and noise [6]. In this paper we propose BWM to assess human face photographs that have noise and high resolution. When we see an image, our attentions are automatically drawn to human faces. In this paper the assessment results based on our natural instinct are discussed.

B. Previous work

Digital cameras became a commodity and our daily lives are filled with digital images. In this environment, image assessment became an interested research field [3][4][5]. However, any study focusing on evaluation of printed photographs has yet to be reported. One study indicated that observers paid more attention to image quality marks [3] but the details of the evaluation were not shown. Therefore, image quality assessment for printed photographs must develop an evaluation method and investigate evaluation tendencies.

In our previous study, subjective assessment BWM in still life photographs achieved a good result [6]. BWM can evaluate images of slight quality differences, but it is impossible to use typical subjective assessment methods on such images. As an example of typical methods, the normalization ranking method [7] is not reproducible because the quality difference of the images is too small. At another example, the Double-Stimulus Impairment Scale (DSIS) method [8], was used to evaluate slightly different images, but it is not a method for evaluating several images. Printed photographs of images taken with digital cameras are often compared to another. Therefore, the evaluation method requires considering the conditions, such as several images and slight differences. BWM in previous study performed well under these conditions.

The results of our previous study also showed the effectiveness of BWM. Furthermore, the surveys found specific evaluation areas regarding noise and resolution. Noise was evaluated in a flat dark area. Resolution was evaluated in an area of fine pattern. In addition it is possible to get reproducibility results when evaluable noise and resolution area exist separately in the same image.

In many cases people take personal images. Humans focus on the faces in these images. This property affects the processes of the image quality evaluation. However, the processes of such images have not been elucidated. The photographs of faces have the same properties as the photographs of still-life. In general, noise of images increases with ISO sensitivity increment and resolution increases with image sensors. In other words noise and resolution are the most important factors when assessing the quality of taken images. Evaluation of the image quality factors and the evaluated areas vary with the amount of noise. Therefore, subjective assessment must consider three things when human face images are evaluated. The first one is to use images taken with varied ISO value. The second one is the investigative processes when evaluating noise and resolution. The last one is the usage of an experimental method such as BWM for evaluating images differences.

This study investigates evaluation processes in human face images. Then subjective assessment using BWM will produce reproducible results for evaluation processes.

This paper is organized, as follows. In Section 2, the subjective assessment materials, the evaluation method and the experiment procedures are explained. In Section 3, the experiment results are presented. In Section 4, the results are analyzed and discussed. In Section 5, we conclude our report.

II. SUBJECTIVE ASSESSMENT METHOD

This section explains necesary elements of subjective asessment. The elements are experimental photographs, evaluation method, and experimental procedure.

A. Photographs used

The photographs for assessment are shown in Figures 1-4. These photographs were named Entrance, Side by Side, Two Rows and Group, respectively. The size of all photographs is 297 ×210 mm (almost letter size) and they are printed with high resolution images of 4.048 × 3,048 pixels. These images were taken with varied ISO values. The ISO values are ISO 100, 200, 400, 800, 1600, 3200, and 6400. The camera was operated in full auto-mode.

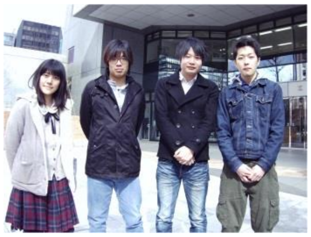


Figure 1. Entrance

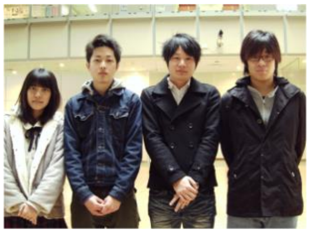


Figure 2. Side by Side

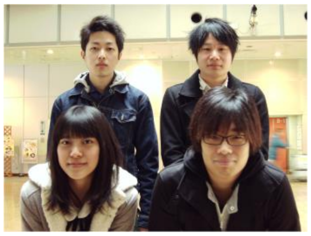


Figure 3. Two Rows

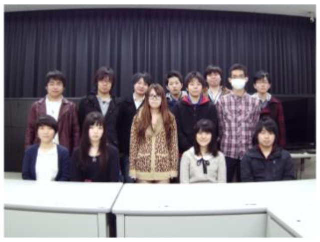


Figure 4. Group

B. BWM

Firstly, observers select the highest and lowest quality images. These images are then disqualified. Then, the process is repeated on the remaining images until there is only one photograph left. Finally, the observers give higher and lower ranks to disqualified photographs, and give middle rank to the last selected photograph.

C. Experimantal procedure

Observers for this study are 21 men and women of 20 years of age. The number of observers was set in accordance with International Telecommunication Union recommendation ITU-R BT.500-13 [8]. The observers evaluate the experimental photographs using the BWM and assign the quality ranks. The ranks are used to get evaluation scores. Range of the scores is from 7 to 1 because the photographs of seven different ISO values are evaluated at one time. The larger scores mean higher quality. The ISO 100 photograph of Figure 3 was excluded from the experimental photographs because it appeared to be severely degraded. Therefore, photographs of Figure 3 have the scores from 6 to 1 according to the image quality.

Observers received a training session because this experiment targets people without any knowledge of images. The training session was used to explain noise and resolution. A still-life photograph that was not evaluated was used for descriptive purpose. Items of the description are determinable noise or resolution areas and strength of the factors. The observers were trained while looking at the still-life photograph. The determinable noise areas and the non-edge flat areas. In the description of the noise, the observers were explained to recognize the presence of noise in various areas of the photograph. In the description of the resolution the observer focused on fine-edged areas after that they were told that non-blurry areas were high resolution.

This experiment focuses on evaluation process of face images. After the experiments observers were questioned about the areas in each image that were interesting in the evaluation aspects. There were two main areas of interest. The first one was the observer's ability to pick out the areas of each image, which were interesting in terms of evaluating the noise and the resolution. The other one was which areas in each photo the observers chose to focus on first.

III. Results of Experiment

Subjective assessment results are shown in Figures 5-8. Diamond points on the scale are the average of the scores and the bars are deviations. The horizontal axis shows the ISO values. The values of standard deviations are represented on the right-hand side of the diamond points. These figures indicate the quality differences in each image. In the Figures 5-8, the quality differences can be guaranteed statistically if there is not overlap of the bar between images. The statistical difference is explained by the probability density function regarding the normal distribution. The state without overlap of the bars represents the sufficiently small probability that the evaluation scores of each image are too similar. The probability is calculated by the probability density function. If the probability is sufficiently small, image quality difference is sufficiently significant. In Figures 5-8, the images without the lowest ISO value indicated quality difference because there were no overlaps of the bars between them. Therefore, this experiment was in obtaining the quality differences.

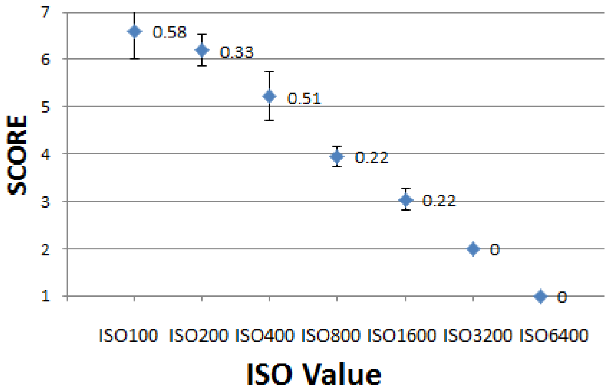


Figure 5. Result of Figure 1

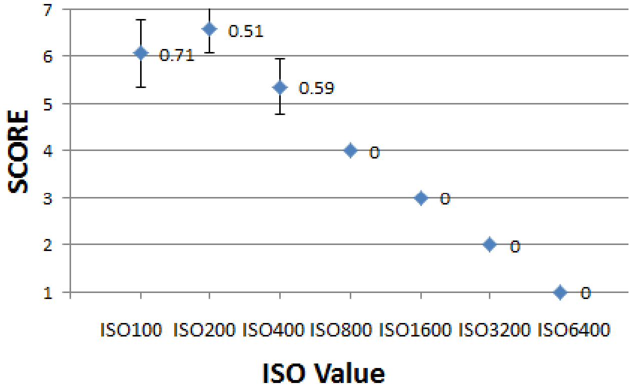


Figure 6. Result of Figure 2

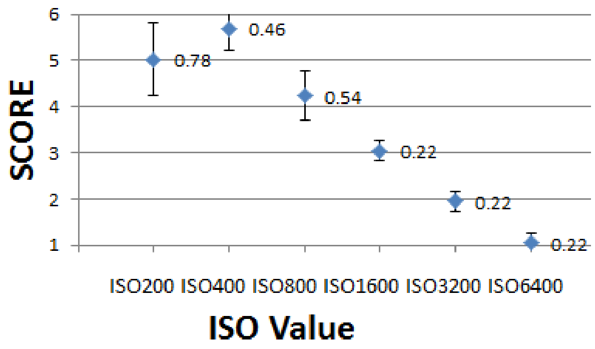


Figure 7. Result of Figure 3

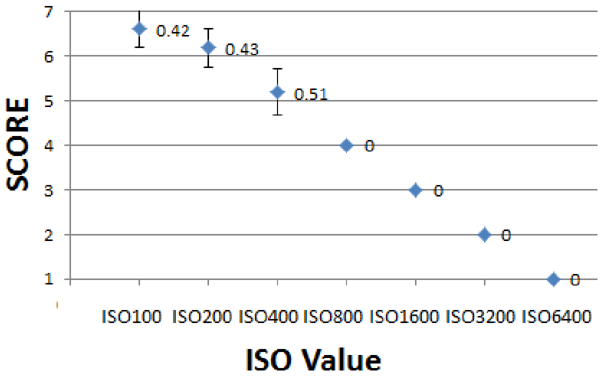


Figure 8. Result of Figure 4

TABLE I. Interesting Areas in terms of Noise and Resolution

Name	*Noise*	*Resolution*
Figure 1	Wall of building, Cheek	Hair, Outline of face
Figure 2	Cheek, Forehead	Hair, Eye
Figure 3	Cheek, Forehead	Hair, Eye
Figure 4	Curtain, Cheek	Hair, Outline of face

TABLE II. Number of People Interested in the Face Areas

Name	*Noise(Unit: people)*	*Resolution(unit: people)*
Figure 1	12	6
Figure 2	21	14
Figure 3	21	19
Figure 4	10	9

The results of questions regarding interesting areas are shown in Table 1 and 2. Table 1 represents the interesting areas for evaluation of noise and resolution. From the areas, specific features regarding evaluation of noise and resolution were acquired. The features of noise evaluation were that the areas which contained flat component. The resolution is evaluated fine-edges in each image. The flat areas are walls of building and cheeks. The fine-edged areas are hair, eye and outline of face. A few observers evaluated the areas other than those listed in Table 1. These areas also had the same features.

Table 2 shows number of the observers evaluating the face areas. A face has hair, cheeks, eyes, etc. The numbers were counted independently. The presence or absence of evaluation in the face areas is shown in Table 2. All observers evaluated noise in the face areas of Figure 2 and 3. However, Figures 1 and 4 were not evaluated in certain area. Therefore, in Figure 2 and 3 existence of the face evaluation was acquired. Although the experimental images had many differences of subject conditions, only Figures 2 and 3 have the common feature. The feature is the larger face than in Figures 1 and 4.

The results regarding the first choice evaluating areas are shown as following. All observers selected the face areas for evaluating at first in Figures 1-4. This result was the same regardless of the areas chosen for evaluating the quality.

The results regarding evaluation noise and resolution were acquired from the questions after the assessments. The results showed that noise becomes visible in the photographs over ISO 800. It was verified such evaluation according to the results that 20 observers evaluated noise in the images.

IV. Discussion

The results are discussed to indicate effectiveness of BWM in this section. In addition they are analyzed to perfect the evaluation processes.

A. *Effectiveness of BWM*

The effectiveness of the BWM is shown by accurate and reproducible results. In Figures 5-8, the evaluation score decreases with increase of ISO value. It means that these experimental results are accurate. The accurate results prove that each photograph is given correct evaluation score. In order to verifying this method, we have to compare the experimental results with general image quality evaluations. In general, the images taken with high ISO value contain high levels of noise. Because the images with high levels of noise are low quality, the images of high ISO values should be evaluated at low scores. In Figures 5-8, these experimental results are consistent with the general image quality evaluations. Therefore, experiment of this study obtained accurate results.

The reproducibility is the most important things in subjective assessments. In Figure 5-8, each mean of scores are separated beyond the deviations. It means that the subjective assessment results have high reproducibility. The images were taken at the same place with camera in auto-mode. Only ISO values are different. Although some images looked similar, observers were able to assess them with reproducible results. It means that BWM can be applicable to the evaluation of images with slight different image quality. The similar assessments for still life with BWM were successful. The difference between still life photographs and personal one is just one thing whether the people were present in them. That difference is the interesting point when it comes to the observers watching the images. If there are men in the images, observers watch human faces. The first areas of focus strongly affect the assessment results. This is according to our experience.

B. *Variations of evaluation areas by face size*

Figures 2 and 3 were evaluated in face areas. These figures had larger face areas than Figures 1 and 4. Therefore, this section will show numerical values of face size to indicate a criterion of evaluation processes. In addition variation of the processes due to changing face size will be discussed.

The size of face area was calculated to acquire numerical criterion in each figure. The size means the ratio of the face area to the entire image. The areas that were used for calculating are shown as square areas in Figure 9. Width of the squares was decided according to the contour lines of cheeks. Height of the squares was selected by distance from the forehead to the chin. These face areas were named Area A, B, C, and D. Number of pixels of the each area was used for calculating the ratio. The calculating results of face size are shown in Table 3. The ratio of Area A means that Figure 1 contains the face areas of 1.0% size. The values of centimeter in Table 3 represent the size of face area in printed photographs. In case of Figure 1, actual size of 1.0% means area of 6.237 cm^2 because the area of photograph in this experiment is 623.7 cm^2.

By these results, existences of face evaluation were shown in areas of 1.5% face size or more. Only Figures 2 and 3 contained such face size. In experimental results, the figures were evaluated in face areas. Therefore, the numeric values are consistent with the experimental results. It means that the numeric value of necessary face size for evaluation is 1.5% or more.

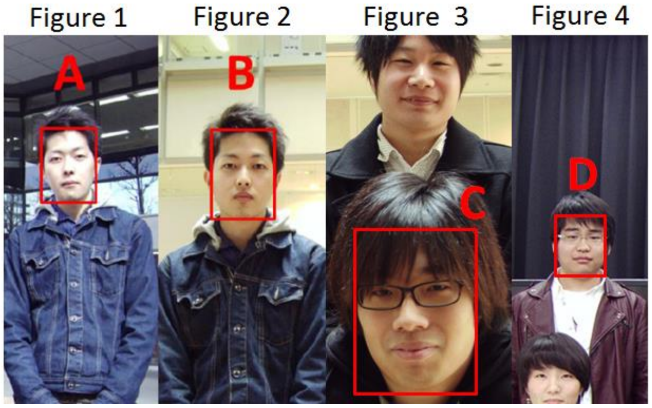


Figure 9. Cut face areas

TABLE III. Ratio of Face area

Name	*Ratio (%)*	*Area (cm2)*
Printed Photograph (297 ×210 mm)	100	623.700
Figure 1: Area A	1.0	6.237
Figure 2: Area B	1.5	9.356
Figure 3: Area C	4.4	27.443
Figure 4: Area D	0.4	2.495

ISBN: 978-1-61208-464-0

In other words the observers evaluate face areas if an image contains face size of 1.5% or more. The face size represents area of 9.356 cm^2 in printed photographs. It means that face area of 9.356 cm^2 in photographs are required for face evaluating. The size of 9.456 cm^2 is area of approximately 3.6 cm × 2.6 cm.

C. Image quality factor by changing physical amount of noise

The experimental results showed that noise areas are evaluated in images above ISO 800. This section confirms the accuracy of these results by calculating a physical amount of noise in the evaluation area.

In general, calculating noise requires flat areas of luminance, but it is difficult to find such areas in image. In this study, areas containing ramp of luminance were selected to compare noise amount with evaluation results. Actually, calculated results matched subjective scores. Figure 10 shows selected areas for noise calculation as the areas in the frames. These areas are 100 × 100 pixels and were selected based on experimental results.

Noise amounts were calculated using (1) and (2) in each ISO value images. The noise amount was required as the decibel (dB) value that each ISO value compared to ISO 100. The large dB value means that the image contains much noise. The equation to get the values is shown as following.

$$\mathrm{MSE_{ISO}} = \frac{1}{N^2}\sum\sum(\mathrm{Y_{ISO}} - \mathrm{AVE_{ISO}})^2 \qquad (1)$$

$$\mathrm{ANS_{ISO}} = 10\log\frac{\mathrm{MSE_{ISO}}}{\mathrm{MSE_{100}}} \qquad (2)$$

There are two calculation processes. They are mean square error (MSE) and logarithm. The equation of MSE is shown in (1). MSE represents the differences between pixel values and mean of the values. The value of MSE is acquired in each ISO value. The dB value is calculated by MSE. The equation is shown in (2).

The calculated noise results amounts are shown in Figure 11-14. The horizontal and vertical axes represent the amount of noise and the means of evaluation scores respectively. ISO value is shown at the top of circle plots. The existence of noise evaluations in each image is shown by the calculated noise results amounts. Confirmation of the existence needs to refer to the relation between the evaluation scores and the noise amounts. In the calculated results if the amount of noise increases with decreasing evaluation score of the image, it indicates such images are evaluated by noise. The reason is that the amount of noise in the evaluation areas affects the evaluation scores. According to Figures 11-14, noise is major factor to determine the image quality over ISO 800. This result matches the observers comments after the assessments of Figures 1-4. In Figure 12, the score of ISO 200 is higher than that of ISO 100. This is caused by the resolution. It means that noise levels of these photos were assessed similar, and that the resolution of the photo of ISO 200 was evaluated higher than that of ISO 100.

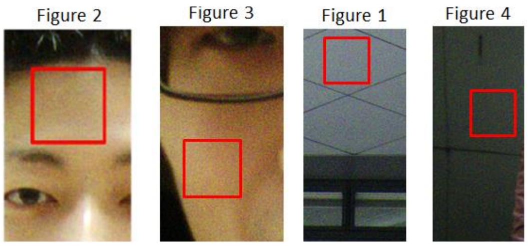


Figure 10. Cut area to calculate noise level

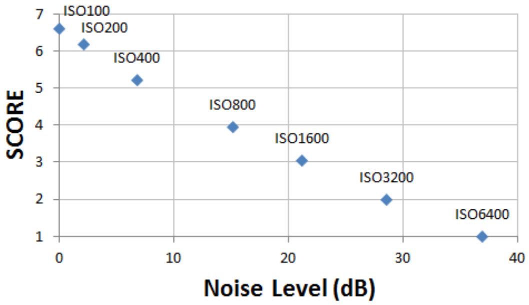


Figure 11. Results of noise calculation: Figure 1

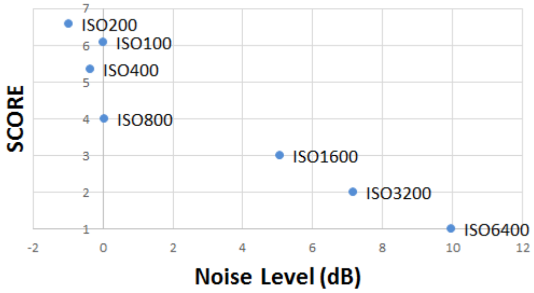


Figure 12. Results of noise calculation: Figure 2

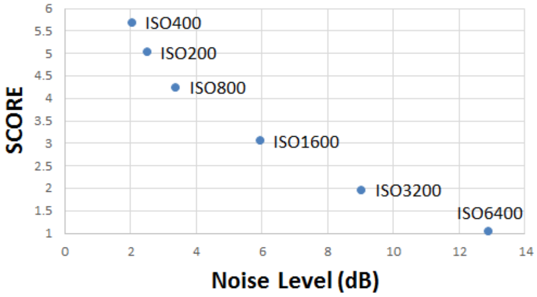


Figure 13. Results of noise calculation: Figure 3

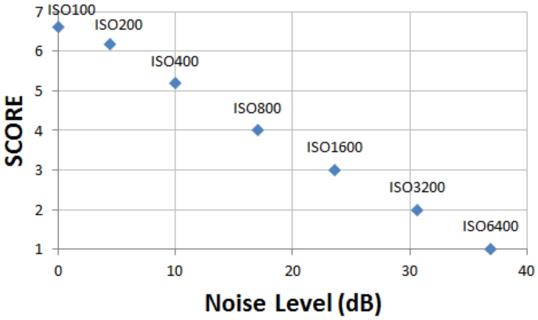


Figure 14. Results of noise calculation: Figure 4

ISBN: 978-1-61208-464-0

Overall results of noise amount are discussed. The scatter points of ISO 800 or greater showed decreasing evaluation scores in accordance with increasing amounts of noise in all images. Furthermore, survey results regarding image quality factors required that ISO 800 or greater images are evaluated by noise. Therefore, these calculated results indicated that there is noise evaluation in ISO 800 or greater images.

D. *Overall evaluation by noise and resolution*

Resolution and noise are high frequency components that appear as texture or noise on images. It is extremely difficult to separate these components using signal processing. However, human vision can evaluate such images by separating them. The experiment of this study considered noise and resolution as the factors of image quality and investigated the evaluation processes of these components by human subjectivity. This section discusses how to evaluate noise and resolution in image quality assessment of human vision.

Contradictory factors (noise/resolution) are evaluated in photographs with different ISO sensitivity value. The overall evaluation process is as follows. First, the observers try to evaluate noise in face area. If they cannot evaluate the face area, they evaluate noise in area other than face. Second, when they cannot evaluate noise, resolution of face area is evaluated. In the same way as the noise, if evaluation of resolution of face is impossible, the observers evaluate areas other than the face. In other words, if they cannot evaluate the face area, the evaluation is done in the area around the face. At evaluating time, noise is preferentially evaluated.

V. Conclusion

In this paper the subjective assessment for noise and resolution in photographs with human faces is proposed. It is our nature that we try to find human faces when we see an image. Because of this, observers try to evaluate the image quality in human face areas. However, it is not easy to evaluate the contradictive factors, noise and resolution in face areas. BWM was applied for observers to recognize the difference in noise levels and resolution. The BWM assessment results are theoretically analyzed and the statistical differences are obtained. BWM is effective when evaluating the contradictive factors and minor level differences. The following facts are also shown in our experiments. Face areas in an image are initially evaluated. If the observers are not able to find image quality differences in the face areas, they shift their attention to the background in order to evaluate noise. In the evaluation, noise preceded the resolution. Furthermore this study indicated that the certain size of face area is required for the face evaluation. The optimal size is 1.5% or more of the entire image.

The results in this paper could contribute to the future digital camera solutions. In the future it will become necessary to evaluate images with changing photography.

References

[1] D. Das, H. J. Mills, and S. Collins, “A wide dynamic range CMOS image sensor with the optimum photoresponse per pixel”, IEEE International Symposium on Circuits and Systems (ISCAS), Rio de Janeiro, Brazil, May 15-18 2011, pp.1560-1563.

[2] ISO 5800:197, “Photography – Colour negative films for still photography – Determination of ISO speed”, November. 1984.

[3] A. E. Boberg, “Subjective vs Objective Image Quality”, XVIIth ISPRS Congress Technical Commission 1: Primary Data Acquisition, August, 1992, pp.73-78.

[4] Y. He, Y. Xuan, W. Chen, and X. Fu, “Subjective Image Quality Assessment: a Method Based on Signal Detection Theory”, IEEE International Conference on Systems, Man and Cybernetics, 2009(SMC 2009), San Antonio, USA, Oct. 2009, pp.4915-4919.

[5] R. K. Mantiuk, A. Tomaszewska, and R. Mantiuk, “Comparison of Four Subjective Methods for Image Quality Assessment”, COMPUTER GRAPHICS FORUM Volume 31, August. 2012 pp.2478-2491.

[6] H. Takeshita and S. Gohshi, “Subjective Assessment for Digital Images”, IEEJ International Workshop on Image Electronics and Visual Computing 2014 (IEVC2014), October 7-10 2014, 3C-3.

[7] T. Fukuda and R. Fukuda, “Guide of Human Engineering: How to scientific sensibility”, 2009, pp.73-123.

[8] ITU-R Recommendation BT.500-13,“Methodology for the subjective assessment of the quality of television pictures”, Jan.2012.

[9] Digital Image Processing Compilation Committee, “Digital Image Processing”, CG-ARTS Society, 2004, pp.79-80.

Object-based Video Coding for Arbitrary Shape by Visual Saliency and Temporal Correlation

Kazuya Ogasawara, Tomo Miyazaki, Yoshihiro Sugaya, and Shinichiro Omachi
Graduate School of Engineering, Tohoku University
Sendai, Japan
email:{oga203, tomo, sugaya}@iic.ecei.tohoku.ac.jp, machi@ecei.tohoku.ac.jp

***Abstract*—This paper addresses a problem of object-based video coding. We propose a video coding method for arbitrary shapes of objects. The proposed method extracts objects on the basis of visual saliency and temporal correlation between frames. Subsequently, we compress the video by changing coding quality for the extracted objects and background regions. The experimental results show that the proposed method can reduce bit rate while preserving target object quality.**

Keywords-arbitrary shape; visual saliency; object-based coding.

I. Introduction

Object-based video coding is effective in many applications, such as video conference and surveillance where we only need regions of interest and it is unnecessary to transmit the entire video. For that reason, object-based coding schemes have been investigated actively. MPEG-4 is a video coding standard for arbitrarily shaped objects. However, this was standardized in 1999, so there is possibility not to preserve important information because compression efficiency of MPEG-4 is lower than current mainstream coding standards. Lan et al. proposed an object-based coding scheme that determines foreground using a depth map and incorporates technologies of MPEG-4 into HEVC [1]. However, it requires a special video camera to capture the depth map. For videos captured by a stationary camera, there are some researches for video surveillance [2] and video conference [3]. In [4], Ng et al. proposed an object-based coding system using multiple video cameras for dynamic image-based representations. However, all of the above methods focus on videos obtained under certain conditions. Many coding schemes have been researched for the purpose of preservation of quality in the area where human tends to perceive. In [5], the foreground is determined on the basis of perception characteristics of human who pay attention on moving objects. In [6], the facial region is set as the foreground, and face parts (e.g., eyes, nose and mouth) are compressed in high quality. However, in these coding systems, the target object is restricted. There are researches which use saliency to determine important regions in videos. The coding method which changes the quality parameter by the macroblock based on saliency is investigated in [7]. However, the method [7] directly uses responses of saliency to compress videos, it does not separate foreground and background. Hence the method doesn't take account of the shape and the contour of the objects.

The contribution of this paper is as follows. We focus on saliency and temporal correlation to extract objects in video. In addition, we propose an arbitrary shape object-based coding scheme which varies the coding quality depending on the foreground and the background. Then, we demonstrate the effectiveness of the proposed method by experimental results.

The outline of this paper is given as follows. In Section II, overview of the proposed method is explained. In Section III, we describe the proposed object extraction method in detail. Afterwards, Section IV proposes an object-based coding system. Finally, Section V shows some experimental results and Section VI concludes this paper.

II. Overview of the Proposed Coding System

The block diagram of the proposed coding system is shown in Figure 1. The encoder extracts visually attractive objects automatically from the input video. Then, we create the foreground video in which background pixel values are equal to 0 and the mask video which consists of mask images. By using a standard video coding method, we encodes these two videos at high quality, and also encode the input video at low quality in order to use it as the background when decoding.

In the decoder, three videos are decoded. Then, the region of foreground is extracted from the high quality object video using the mask. Similarly, the region of background is extracted from the low quality entire video using the mask and the synthesis video is reproduced by combining the foreground and the background.

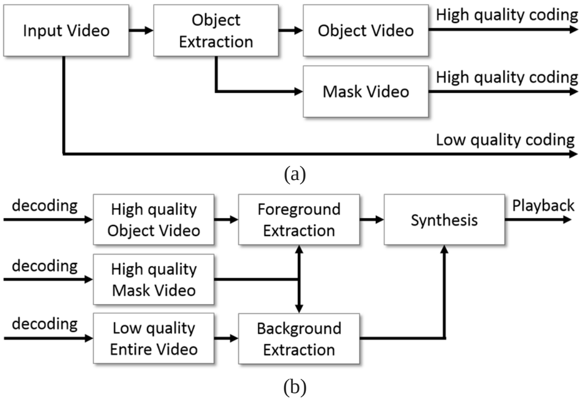


Figure 1. Block diagram of the proposed coding system: (a)Encoder (b) Decoder

III. VISUALLY ATTRACTIVE OBJECT EXTRACTION

This process extracts visually attractive objects automatically. From the input video frames, key frames are chosen at a constant interval. In each key frame, we create a mask which shows the target object area using a saliency map [8] and GrabCut [9]. In other frames, we create it by moving the mask from the key frame to subsequent frames using optical flow. It reduces processing time and keeps the shape of the object among frames not to create masks for each frame from scratch. Furthermore, in order to improve the accuracy of object extraction for key frames, we compare the created mask with a predicted mask based on the mask of the preceding key frame, and create a mask again in a different setting if the difference is large.

A. Key Frame Mask Creation

1) Pre-Processing: Pre-processing is performed to improve the accuracy of mask creation. First, the size of input frames is reduced to make the subsequent process easier. Second, smoothing with Gaussian filter is performed to reduce the high-frequency components which have bad influence on segmentation.

2) Mask Creation: This process estimates the target object area and creates a mask of the objects. We use the saliency map proposed in [8] to estimate it. The saliency map shows the degree of visual attention. In the saliency map, large values represent locations where human will pay attention. Examples of an input frame and its saliency map are shown in Figure 2.

To create a highly precise mask, we use the GrabCut algorithm. The GrabCut is a graph-based two-class segmentation method. In the algorithm, the input image is expressed as a weighted graph based on similarity to samples of foreground and background given by the user and color difference of adjacent pixels. Then, segmentation is conducted by finding the minimum cut to divide the graph into two subgraphs. The saliency map does not always have uniform saliency in the same object, and there are large saliency pixels in background. Therefore, it is difficult to create a mask using the saliency map only. The proposed method realizes to create a highly precise mask by combining the saliency map and the GrabCut. In order to assign sample pixels, a label mask is created. The label mask consists of four values: BGD (to be a background sample pixel), FGD (to be a foreground sample pixel), PR_BGD (to be probably background pixel), and PR_FGD (to be probably foreground pixel). The PR_BGD and PR_FGD pixels are estimated by calculation based on the FGD and BGD pixels. We create the label mask automatically based on the saliency map. First, we hypothesize that humans tend to pay attention to the center of the screen rather than the edge of the screen. Based on this hypothesis, 15 pixels from the border of the image are set as BGD pixels regardless of its saliency. The other pixels in the inside of the screen are set as some label based on the saliency map. Specifically, the high saliency pixel becomes FGD pixel, the middle saliency pixel becomes PR_FGD pixel, and low saliency pixel becomes PR_BGD pixel. Figure 3 shows the label mask. Then, the GrabCut using the label mask creates a mask which shows the area of the visually attractive object in the frame.

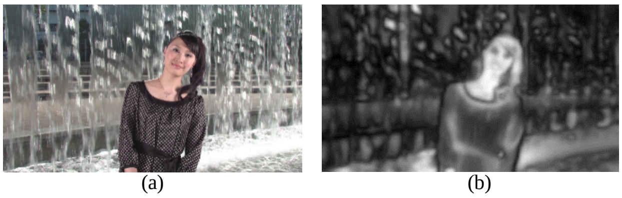

(a) (b)

Figure 2. A saliency map: (a) Input frame "Fountain(Chromakey)" (b) Saliency map

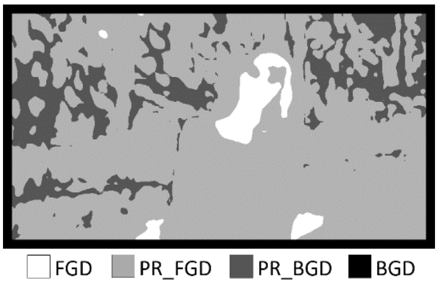


Figure 3. The label mask created based on the saliency map

3) Post-Processing: Two types of post-processings correct misjudged pixels. In the mask created by using the GrabCut, some background pixels may be judged as foreground by mistake. Therefore, we do labeling process to the mask and judge small foreground labels as background, the size of which is smaller than 0.5% of the frame size. Likewise, some foreground pixels may be judged as background by mistake. Therefore, we change the background pixels surrounded by foreground pixels to foreground. This process fills holes of the foreground and improves the mask.

4) Mask evaluation considering temporal correlation: For more accurate mask creation, we do comparative evaluation of the mask that takes advantage of temporal correlation, which is a characteristic of video. Video is agregation of continuously captured images, so it is rare that the contents vary greatly between frames. Therefore, it is not desirable that the shape of the mask varies greatly between frames. However, there is a possibility that the shape of the mask varies greatly when doing object extraction using information about the target frame only.

We compare the created mask and the predicted mask, which is predicted from the mask created in the previous key frame. We use the optical flow to create the predicted mask. First, we calculate the optical flow between continuous key frames based on the method of Gunnar Farneback [10]. Second, we do the labeling process to the mask, and calculate the average vector of the optical flow of each label in the foreground area. Then, we create the predicted mask by moving the foreground area for the average vector of each label. The number of pixels where the predicted mask does not overlap with the mask created using the GrabCut is found by taking exclusive OR of them. If the number becomes more than 10% of the number of all pixels in the frame, we conduct the following process for the mismatched pixels. When the pixel of the predicted mask is the background pixel, we set the pixel of the label mask as PR_BGD. Similarly, when the pixel of the predicted mask is the foreground pixel, the pixel of the label mask is set as PR_FGD. Then, we conduct the GrabCut algorithm again using the renewed label mask.

B. *Mask Creation in the Non-Key Frame*

For the non-key frame, the mask is created by moving the previous mask created already using optical flow. This process is generally the same as creating the predicted mask for comparative evaluation, but uses the optical flow between continuous frames. It reduces the shape change of the foreground to move the mask based on the optical flow.

C. *Object Extraction*

In this process, we enlarge the masks to the size of the input frame and extract the foreground from the input frame based on the mask. Then, the foreground video, of which pixel values in the background are 0, is created. Figure 4 shows the foreground frame. We create the mask video by aggregating the masks. The mask video is utilized to determine the background at the time of decoding.

IV. CODEC SYSTEM

A. *Encoder*

In the proposed method, the encoder realizes preserving important information and reducing unnecessary information by changing the coding quality in H.264 according to the type of video. The foreground video is coded in high quality, so that information of the target object is preserved. Because the value of all pixels of the background area is zero, it is possible to reduce the data size with high quality. In addition, the mask video used in decoding is coded in high quality, so that boundary between the foreground and the background is preserved. Furthermore, the input video which is the subject of object extraction is coded in low quality, and the data size is reduced significantly. This entire coded video is used as the background at the time of decoding.

B. *Decoder*

This process decodes three kinds of coded videos based on H.264 at first. Next, we extract the foreground area from the object video coded in high quality based on the mask. Similarly, we extract the background area from the entire video coded in low quality based on the mask. Then, we synthesize the foreground and the background. Eventually, we can obtain the synthesis video which preserve the quality of the foreground and reduce the quality of the background because the coding quality is different in the foreground and the background. Since it is not the block based quality control, we can handle arbitrary shape objects. Figure 5 shows the synthesis frame.

V. EXPERIMENTS

In order to confirm the effectiveness of the proposed method, we conducted a comparative experiment with H.264. We used three video sequences at FHD resolution (1920×1080): “Fountain(chromakey)”, “Fountain(dolly)” and “Truck Train” included in the ITE/ARIB Hi-Vision Test Sequence 2nd Edition [11], shown in Figure 1 (a) and Figure 6. In the proposed method, we set the key frames at intervals of three frames, and the same low-quality entire video is used for background regardless of the bit rate.

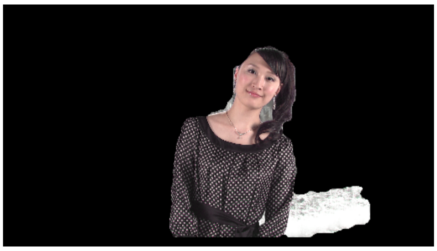


Figure 4. The foreground frame

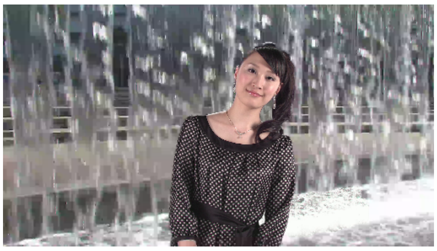


Figure 5. The synthesis frame

A. *Rate-Distortion Performance Evaluation*

In this subsection, we evaluated the coding performance by rate-distortion curves, which indicates a relationship between the bit rate and PSNR. Figure 7 shows the rate-distortion curve in whole and foreground regions.

In the proposed method, the image quality of the foreground was good, but that of the background was significantly degraded. Therefore, PSNR on the entire area became low values. In contrast, the video coding standards, such as H.264 utilize rate-distortion optimization to choose the partition manner and the coding mode, so PSNR is optimized. For that reason, PSNR of H.264 became larger than that of the proposed method.

On the other hand, we confirmed that PSNR on the foreground of the proposed method became large than that of the H.264 at various bit rates. This is because that the proposed method reduces the bits about the background significantly. This result shows that the proposed method can reduce more bit rate than H.264 when the image quality of the foreground is compressed to the same degree between the proposed method and H.264. However, as for “Truck Train”, the PSNR on the foreground of the proposed method became lower than that of H.264 at a low bit rate. The background of “Truck Train” is not complicated. Therefore, the proposed method could not have enough effects. In addition, inaccuracy of object extraction affected the results.

B. *Subjective Quality Evaluation*

In this subsection, we evaluate subjective quality of the foreground by measuring mean opinion scores (MOS). In this experiment, all the video sequences compressed by the proposed method and H.264 at different bit rates were displayed in a random order, MOS is ranging 1 to 10. We had 13 participants involved in this experiment.

Figure 8 shows the MOS values. As for “Fountain

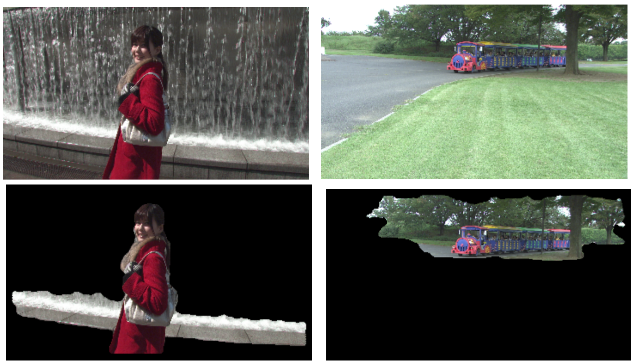

(a) (b)

Figure 6. Top row: Original video sequences Bottom row: Object extraction results (a)”Fountain(dolly)” (b)”Truck Train”

(chromakey)” and “Fountain(dolly)”, the MOS values of the proposed method are larger than H.264 at any bit rate. However, as for “Truck Train”, the MOS value of the proposed method is smaller especially at a low bit rate. This result could be due to boundary flicker caused by inaccuracy of synthesis of foreground and background.

VI. Conclusion

In this paper, we proposed an object-based coding system that extracts visually attractive object with arbitrary shape and preserves the information of the object. The region of interest is estimated with the saliency map, and the objects are extracted using the GrabCut.

The experimental results show that the proposed method realizes both preserving the image quality of the target objects and reducing the bit rate. However, there is a possibility of reducing important information greatly if objects are not extracted correctly. Therefore, it is necessary to investigate a more accurate object extraction technique. In addition, it is very complicated to hold the foreground video, the mask video and the entire video for background. This reason is that we use the video coding standard, which is not object-based. Therefore, it needs to investigate an object-based encoder.

Acknowledgment

This work was partially supported by JSPS KAKENHI Grand Number 26540077.

References

[1] C. Lan, J. Xu, and F. Wu, “Object-based coding for Kinect depth and color videos,” Proceedings of the IEEE Conference on Visual Communications and Image Processing, San Diego, 2012, pp. 1-6.

[2] R. V. Babu and A. Makur, ”Object-based Surveillance Video Compression using Foreground Motion Compensation,” Proc. ICARCV, Singapore, 2006, pp. 1-6.

[3] Y. Li, X. Tao, and J. Lu, ”Hybrid model-and-object-based real-time conversational video coding,” Signal Processing: Image Communication, vol.35, 2015, pp. 9-19.

[4] K. T. Ng, Q. Wu, S. C. Chan, and H. Y. Shum, ”Object Based Coding for Plenoptic Videos,” IEEE Trans. Circuits and Syst. Video Technol, vol.20, no.4, 2010, pp. 548-562.

[5] M. Bosch, F. Zhu, and E. J. Delp, ”Video coding using motion classification,” Proceedings of the IEEE International Conference on Image Processing, San Diego, CA, 2008, pp. 1588-1591.

[6] M. Xu, X. Deng, S. Li, and Z. Wang, ”Region-of-Interest Based Conversational HEVC Coding with Hierarchical Perception Model of Face,” IEEE Journal of Selected Topics in Signal Processing, vol.8, 2014, pp. 475-489.

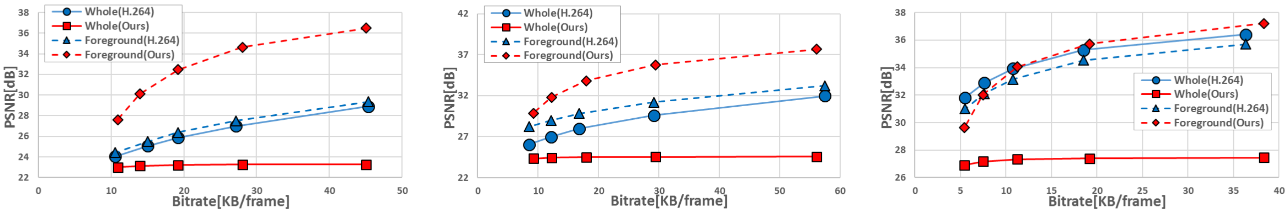
[7] H. Hadizadeh and I. V. Bajic, ”Saliency-Aware Video Compression,” IEEE Trans. on Image Processing, vol.23, Issue.1, 2014, pp. 19-33.

[8] L. Itti, C. Koch, and E. Niebur, “A Model of Saliency-based Visual Attention for Ra-pid Scene Analysis,” IEEE Trans. on PAMI, vol. 20, no.11, 1998, pp. 1254-1259.

[9] C. Rother, V. Kolmogorov, and A. Blake, “GrabCut: Interactive Foreground Extra-ction using Iterated Graph Cuts,” Proc. on ACM SIGGRAPH, 2004, pp. 309-314.

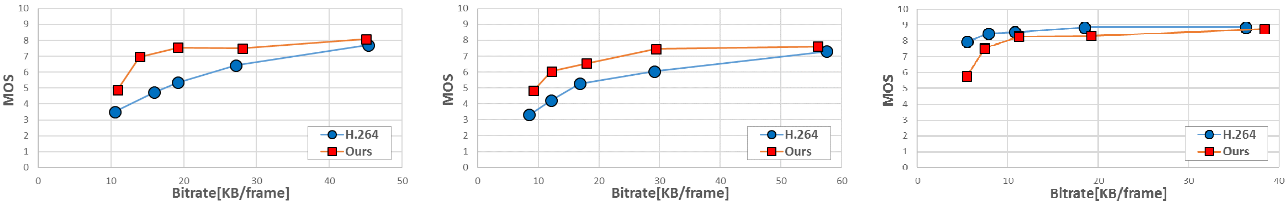
[10] G. Farneback, ”Two-Frame Motion Estimation Based on Polynomial Expansion,” Proc. of 13th Scandinavian Conference on Image Analysis, SCIA, 2003, pp.363-370.

[11] ITE/ARIB Hi-Vision Test Sequence 2nd Edition Reference Manual, 2009.



(a) (b) (c)

Figure 7. Rate-distortion curve: (a) “Fountain(chromakey)” (b) “Fountain(dolly)” (c) “Truck Train”



(a) (b) (c)

Figure 8. MOS value: (a) “Fountain(chromakey)” (b) “Fountain(dolly)” (c) “Truck Train”

Procedural 3D Urban Content Generation in Simulation and Games

Noura El Haje
Jean-Pierre Jessel
Institut de Recherche en Informatique
de Toulouse
Paul Sabatier University
Toulouse, France 31400
Email: Noura.El-Haje@irit.fr
Email: Jean-Pierre.Jessel@irit.fr

***Abstract*—As virtual urban environment generation becomes a widespread research topic, the need to create enriched urban worlds, which group both the geometric and semantic properties has become a necessity, the purpose being to obtain interactive, adaptive and consistent world entities. This paper suggests a representational model of urban environment generation based on both geometric and semantic knowledge, the future work being to create a modeling tool of a 3D urban environment, which stores the semantic information and allows the final non expert user to acquire information about the environment before and after its creation and to explore this information in an interactive way.**

Keywords–3D Content; Semantic-Based; Enriched Environment; Web Content.

I. Introduction

Virtual environment generation techniques and modelers have become a powerful and efficient way for creating plausible 3D environments destined for visualization, simulation or serious games. The automatic generation of these virtual environments has also reduced the time and cost charges of the manual generation, which can be very laborious and repetitive for games developers. Although works were established on visualization and generation tools concepts, there is not so far an intuitive generation solution that makes use of the semantics in a way that allows the final user not only to acquire basic knowledge about the world, but to deal with the semantics at a higher level.

In Section 2 of our paper, we will briefly elaborate the most relevant urban generation techniques with their advantages and limitations. We will then describe in Section 3 our representational model called UDG (Urban Data Generation) along with the standards and techniques used to create this model. Section 4 develops the mecanism of the representational model and discusses what needs to be done and the challenges behind information retrieval from the city generated before concluding our paper in Section 5.

II. Related Works

This section develops the most relevant works to our research focus, mainly with the preliminary works on procedural urban models generation to the most recent approaches in this domain. The semantic-based generation topic, along with its advantages and challenges is also elaborated.

A. Urban Generation

A common approach for procedurally generating cities is to start from a dense road network and identify the polygonal regions enclosed by streets. Then, the subdivision of these regions results in lots, which are populated with buildings in two ways: either the lot shape is directly used as a building footprint or the building footprint is fitted on the lot. Finally, by simply extruding the footprint to a random height, it becomes possible to generate quite complex models [1]. Some approaches like split grammar [2] and shape grammar [3] have been used for generating several kinds of pattern structures by executing constructive solid geometry operations on a set of components selected by query mechanisms. An approach that also retains the advantages of grammar-based solutions is the procedural content graphs [4]. The content generation procedures are specified by a generic approach, which gives a richer and a more expressive design specification than that offered by other grammar and graph-based approaches.

A more recent concept in the domain of procedural modeling, the inverse procedural modeling, was discussed by Musialski et al. [5]. It aims at discovering both the parametrized grammar rules and the parameter values that yield to a prespecified output when applied in a particular sequence. The most important advantage of the inverse approach is that it leaves the designer entirely out of the process: the Procedural modeling is opaque so the designer does not need to know about it at all [6]. Another work on the topic of inverse modeling was established by Vanegas et al. [7] who propose a framework that enables intuitive high-level control to an existing urban procedural modeling by interactively edit urban model. Their system is capable of discovering how to alter the parameters of the urban procedural model so as to produce the desired 3D output. The limitation of this method is the lack of accuracy of their engine with parameters increase, especially when a large number of dependant parameters are included.

Most of the approaches presented above are based on traditional procedural modeling, which was and is still considered a significant contribution for the computer graphics community. However, the major drawback of this approach is that the number of rules to be defined is quickly increasing with the size of the environment and the level of detail (LOD), which becomes difficult to control and maintain.

B. Semantic-Based Generation

The majority of procedural methods are specialized in generating one specific type of content or feature. To be really useful in the design of complete virtual worlds, heterogeneous data needs to be assembled, explored and integrated in a modeling tool. The advantage of making use of heterogeneous data for urban environment generation is that it allows designers to focus on high level concepts in the world design rather than on the model level [8]. In this direction, the authors described their framework (SketchaWorld) based on a declarative modeling approach and which combines the strength of manual and procedural modeling by allowing designers to state their intent using simple, high-level constructs. Their declarative approach builds upon established research results on parameterized procedural generation, constraint solving and semantic modeling in order to automatically tarnslate statements into a matching 3D virtual world. The consistency of this world is automatically maintained using a semantically rich model of all its features and their relations, analogous to the automatic maintenance of interior scenes based on object semantics.

The lack of rich information suitable for consumption by the game artificial intelligence (AI) was exposed in [9] where the solution suggested the improvement of the embedded information contained in immersive game worlds by focusing on the symbolic annotations of environmental elements. Some authors [10] focused on applying the semantics for managing digital representations of buildings for architectural applications and analyse their transformations over time. Their description model defines three levels of the building morphology: the semantic, the structure and the representation one and the temporal dimension is joined to these three levels. Also in the context of the city change over time, an interactive agent-based behavioral system was elaborated in [11]. This system is linked to the geometrical modeling in order to create plausible urban models. Their platform (NUBES) focuses on the definition of an informative system on an architectural scale, which exploits the relations between the 3D representation of the building and heterogeneous information coming from various fields (technical, documentary or historical). It aims at organizing multiple representations (and associated information) around a model of semantic description with the aim of defining a system for the multi-field observation of historic buildings.

III. Our Proposed Representational Model

This work in progress suggests a representational model of urban data that associates semantic information to 3D urban data. This model will be implemented in a modeling tool so that the final non expert user could visualize the data and explore the semantics in order to create his own modified urban environment. Another usability of the semantics will be to make the environment architecturally more complex by adding texture and geometric details. The previous works explained above showed how buildings and complete cities could be generated based on heightmaps and procedural rules.

A. Standards and Tools

Our proposed model called urban data generation (UDG) is built around some standards and tools for a complete environment representation. First, we focused on extracting geospatial data layers of cities from the openly licensed map, OpenStreetMap which is a viable alternative to commercially offered data sources. The exported format, the OpenStreetMap (OSM) format is coded in extensible markup language (XML) and contains geographic data in a structured, ordered manner.

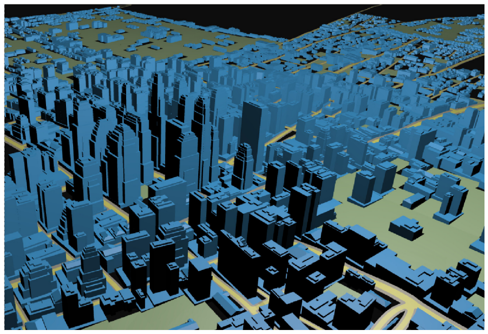


Figure 1. X3D file display for an intermediate city size in html page

In order to represent the OSM file in 3D, we focused on the Open-Source web graphics library (WebGL) framework and in particular the document object model (X3DOM) for two main reasons: First it's an ISO standard file format which represents 3D graphics using XML syntax, which makes it easy to undestand and modify. Second, its runtime environment manages interoperation between the extensible 3d (X3D) browser and host application for purposes like file delivery, hyperlinking, page integration and external programmatic access. Finally, this format is supported by many 3D modeling softwares such as Blender, Meshlab and 3Ds max(via a plugin) which enables the environment to be externally edited by the designers. Figure 1 shows an example of 3D urban data displayed as a webpage that takes an urban file in the open street map (OSM) format and displays it as an extensible 3d (X3D) file.

The X3D file is then imported into the game engine Unity3D using the X3D plugin for Unity. The use of Unity3D in the development allows a better control of the city layers and an easier fusion of the semantic information with the geometric one. The constraints of only using the X3D format to visualize the city is the lack of the semantic information and thus a lack of ”character” to the buildings. To solve this problem we needed to enrich the extensible 3d (X3D) file by adopting two approaches: 1- adding metadata tags inside the extensible 3D (X3D) file and 2- defining relations between layers models.

The geographic markup language (CityGML) model was our inspiration for enriching the world with metadata. CityGML is a common information model and an XML-based encoding that describes 3D objects with respect to their geometry, topology, semantics and appearance. It also defines five levels of details (LODs). The LOD concept in CityGML is different from the one in computer graphics since it denotes the model spatio-semantic adherence to its real-world counterpart [12].

Buildings at LOD0 in CityGML can be represented in two ways: a footprint and a roof edge (in general). Both the LODs representation of a building footprint and a roof edge have to be a horizontal surface pursuant with CityGML specifications. If a footprint is in reality situated on a slope then the lowest

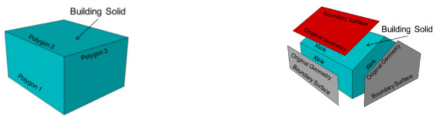


Figure 2. Different LODs of a building taken from CityGML specifications

value has to be used (as specified in CityGML). Figure 2 shows the difference between LODs. Left: and LOD1 solid without surfaces and right: an LOD2 solid with accompanying modelled surfaces.

B. Our Representational Model Description

Our interface of city generation is given some parameters as input. This is the default scenario which is parametrized by the user. Some of these parameters are : the city type (large city, village, small town...), by the sea or by a river and the society type (isolated, mixed...). Our environment consists of three generators: the terrain generator, the street generator and the buildings generator.

1- Terrain Generator: The terrain is generated using the ”CoherentNoise” library, a library for Unity3D that allows the use of noise of all kinds. The library has the ability to modify and combine different noises together, all with a simple and clean object-oriented API. CoherentNoise library allows for unlimited possibilities.In most basic form, noise function can be used to distort some pattern - that may be itself generated procedurally or made by hand. Noise could be directly added to the pattern, or noise values are used to perturb the pattern. Perturbation of function is achieved by adding noise value to the function input. Figure 3 shows the effect of addition and perturbation on a 2D pattern and the terrain sample using the coherent noise function.

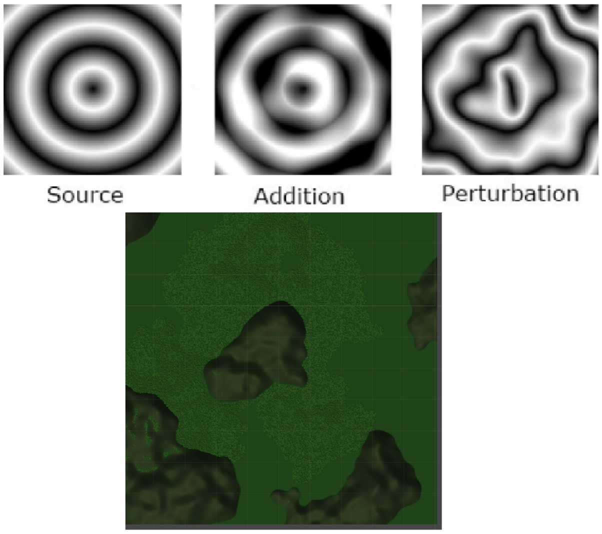


Figure 3. Effect of adding noise to a pattern and the sample terrain we generated using Coherent Noise

2- Street Generator: To create street patterns, we used the famous Voronoi Diagram technique. This technique creates cell-like or network-like patterns. A diagram is based on an (infinite) number of control points, that are randomly distributed in space. Such distribution is not completely random, but it is enough for most practical applications. Base classes for Voronoi diagrams determine distances to three closest control points and this allows to make own diagrams by combining these distances in creative ways. All Voronoi diagrams have a 2D variant. These variants are exactly the same as their 3D counterparts, except they ignore the Z- coordinate. This makes them considerably faster (as less control points must be considered). Figure 4 explains the concept of Voronoi and a street pattern we generated by this method.

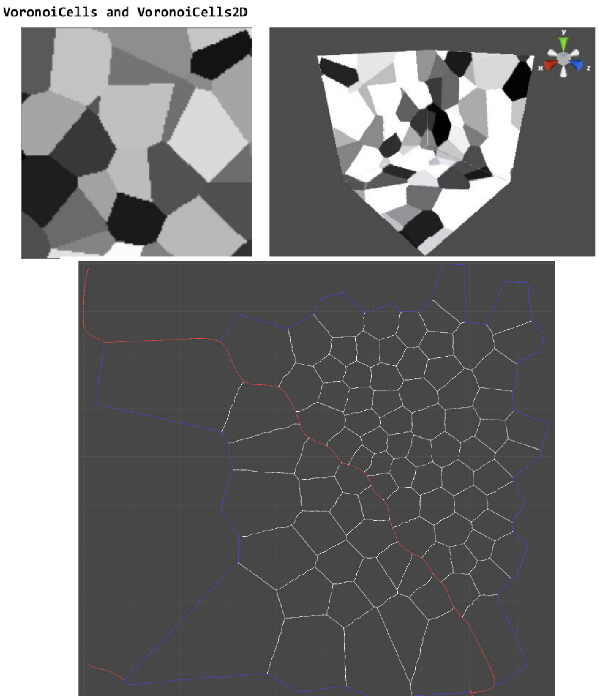


Figure 4. Voronoi graph and streets generated using this technique

3- Building Generator: As we explained before, we are using X3D format to generate the buildings in our environment. The drawback of the X3D is its lack of high level semantic information. The information regarding the buildings is only limited to basic internal properties such as placement, size and material color. Therefore, we resorted to CityGML, a standard that provides five standard LODs: LOD0 is a 2D footprint, LOD1 is a block model obtained with extrusion, LOD2 is an upgrade of the former with simple roof structures and semantically enriched boundary surfaces, LOD3 are architecturally detailed models with fenestration, and LOD4 contains interior [13].

IV. Framework Mecanism and Discussion

The mecanism of our representational model UDG with the different blocks is shown in Figure 5 and will be integrated in a modeling interface largely dependant on the semantics. The modeling interface takes as input the default scenario that gives specification about the city from the user and a default X3D

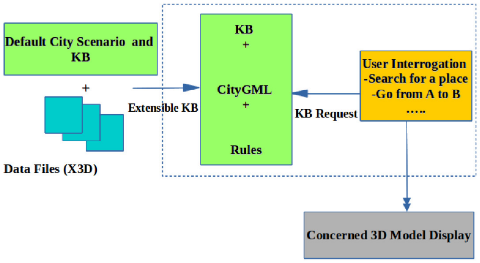


Figure 5. Representational model of the city generation

file. The user using parameters from the terrain generator, the street generator and the buildings generator explained above respectively allows adding more visual details to the city and thus enriching the X3D file. The knowledge base makes use of the CityGML nodes and should be automatically updated. Up to now, the visualization is limited to a general level of detail and the visualization and manipulation of buildings interior is not to be considered at this stage of our work.

After the city generation, the user shoud interact with the city by launching some sort of a query (for example looking for a building that have more than 3 stories or how to go from point A to point B). The knowledge interpretation and its visibility in front of the user is an aspect we are currently investigating. It could be satisfied graphically (by displaying tags on the buildings according to their types or characteristics), or visualizing the acceptance area depending on the user request.

To summarize what we explained before, this is what needs to be done: First, accomplish the development of the modeling interface which receives as input the extensible 3D (X3D) file and displays it. Second, we need to define the interaction types and ways that the user could have with the city generated, which is important for our system functionality validation. And third, rules and constraints should be specified to maintain the world consistency while paying attention to not over-constrain the system, which could lead to unrealistic urban representation.

V. Conclusion and Future Work

We have introduced an approach to combine the geometric urban data with semantic information, both high and low level. The implementation task is in progress and aims at reaching a complete modeling interface for city creation based on many standards, tools and models that make use of the world knowledge. At this moment, the process to provide semantic contents is manually established. We plan at a later stage to automate the generation process of the semantic database.

As a future perspective, semantics could be also interesting in improving the geometric models by adding more architectural details and textures. Our models are basic geometric shapes extruded or refined with no accessories such as windows, roofs or balconies. The semantics could therefore make the environment visually more appealing to the user by allowing various LODs. Finally, as we have seen from the related works on the different levels of semantics, there are promising uses for semantic information in virtual games worlds and in the near future, we expect increasing effort to be made in the merging process of different data type for virtual worlds creation destined for simulation and games.

References

[1] S. Greuter, J. Parker, N. Stewart, and G. Leach, “ Real-time procedural generation of ‘pseudo infinite’ cities,” Proceedings of the 1st international conference on Computer graphics and interactive techniques in Australasia and South East Asia 2003, NY, USA., pp.: 87-ff http://doi.acm.org/10.1145/604471.604490

[2] L. LeBlanc, J. Houle, and P. Pulin, “ Component-Based Modeling of Complete Buildings,” Proceedings of the Graphics Interface 2011, May 25-27 Canada, pp.: 87-94 http://doi.acm.org/10.1145/237170.23719

[3] M. Larive and V. Gaildrat, “Wall Grammar for Building Generation,” Proceedings of the 4th International Conference on Computer Graphics and Interactive Techniques in Australasia and South-East Asia 2006, Canada, pp.: 429-437 http://doi.acm.org/10.1145/237170.23719

[4] S. Pedro Brandao and E Eisemann, “Procedural Content Graphs for Urban Modeling,” International Journal of Computer Games Technology ,volume 2015, Article ID 808904, 15 pages http://dx.doi.org/10.1155/2015/808904

[5] P. Musialski and M. Wimmer, “Inverse Procedural Methods for Urban Models,” Eurographics Workshop on Urban Data Modeling and Visualization, May 2013, pp.: 31-32, Editors V.Tourre and G.Besuievsky

[6] R.M. Smelik, T.Tutenel, R. Bidarra, and B. Bedrich, “A Survey on Procedural Modeling for Virtual Worlds,” Computer Graphics Forum 2014, Volume 33, issue 6, pp.: 31-50 http://dx.doi.org/10.1111/cgf.12276

[7] C. Vanegas, I. Garcia-Dorado, D. Aliaga, B. Benes, and P. Waddell, “Inverse Design of Urban Procedural Models,” ACM Trans. Graph.31, November 2012, Article 168, 11 pages http://dx.doi.org/10.1145/2366145.2366187

[8] R.M. Smelik, B.Tutenel, K.J. De Kraker, and R. Bidarra, “A Declarative Approach to Procedural Modeling of Virtual Worlds ,” Computers and Graphics 35, 2011, pp.: 352-363 http://dx.doi.org/10.1016/j.cag.2010.11.011

[9] G.M. Youngblood, W.P. Frederic, D. Heckel, H. Hale, and P.N. Dixit, “Embedding Information into Game Worlds to Improve Interactive Intelligence,” Artificial Intelligence for Computer Games, 02 February 2011, pp.: 31-53

[10] L. De Luca, C.Busayarat, C. Stephani, P. Veron, and M. Florenzano, “A Semantic-Based Platform for the Digital Analysis of Cultural Heritage,” Computers and Graphics 35, Issue 2, April 2011, pp.: 227-241 http://doi.acm.org/10.1016/j.cag.2010.11.009

[11] C. Vanegas, D. Aliaga, B. Benes, and P. Waddell, “Interactive Design of Urban Spaces Using Geometrical and Behavioral Modeling,” ACM Siggraph Asia 2009, ACM NewYork, USA, Article 111, 2009, 10 pages http://doi.acm.org /10.1145/1661412.1618457

[12] F. Biljecki, H. Ledoux, J. Stoter, and J. Zhao, “ Formalisation of the level of detail in 3D city modelling,” Computers, Environment and Urban Systems 48, November 2014, pp.: 1-15

[13] T.H. Kolbe, “ Representing and exchanging 3D city models with CityGML,” In 3D Geo-Information Sciences, Zlatanova S., Lee J., (Eds.). Springer Berlin Heidelberg, 2009, pp. 15-31. 2

Development of Soft Skin of Digital Hand in Real Time Operation

Hiroshi Hashimoto
Industrial Technology Graduate Course
Advanced Institute of Industrial Technology
Tokyo, Japan
hashimoto@aiit.ac.jp

Kaoru Mitsuhashi
Department of Mechanical Engineering
Tokyo University of Technology
Tokyo, Japan
mitsuhashi@stf.teu.ac.jp

***Abstract*—This paper presents a development of soft skin of a digital hand which mimics human hand and is able to show dexterous operation of an object in real time. In the operation, soft skin as shown in human skin plays an important role in the contact with objects. To develop it, design methods of the soft skin as a model of human skin and of enabling real time operation are described.**

Keywords-digital hand; soft skin; real time operation.

I. Introduction

This paper presents a development of soft skin of a digital hand which mimics human hand and is able to show dexterous operation of an object in real time.

Human hand performs various difficult tasks in daily life and shows dexterous operation to use tools or equipment as object, because it has numerous degree of freedom (DoFs) of finger joints more than 22 DoFs [1][2]. There are many types of grasp such as power grasps, precision grasps and miscellaneous grasps, and each types is also divided into many various hand postures [3]. These hand postures can be made by the hand's DoFs; basically the posture of holding and arch ensure the various hand posture. However, a study on dynamical operation of hand grasping objects has not been made in the field of anatomy, but only on grasping which shows static situation to fix objects.

In the related studies on Computer Graphics (CG), considering muscular, freedom of joints or tendons, a precise digital hand to mimic human hand has been tried to be made [4]-[7]. The purpose of these studies are merely to simulate hand motion or evaluate product designs when it grasps an object. They conduct only static grasping, do not show dynamic operation.

The dynamic operation of the digital hand has been slightly considered in [8][9]. In the researches, the body of the digital hand was made of rigid body. On the other hand, human hand is covered with soft skin which is deformable. While hand operates an object, the contact region between soft skin and the object is area but not point as seen in rigid skin, so the dynamic relationship on the contact region becomes complex. This means the real time operation of the digital hand requires numerous computational load and is hardly realized. Furthermore, manipulating the digital hand dynamically has numerous patterns of the digital hand posture for each operation cases. This leads to that the programming to realize all the patterns of hand motion by using a certain computer language is very troublesome.

This paper proposes a novel design method of making soft skin to be suitable for real time operation of the digital hand, and this is an unprecedented study.

The platform of the system is on Panda3D [10], and the software to make soft skin is Blender [11] which fits for both Panda3D and Python language. To realize the real time operation of the digital hand, Bullet Physics which is a physics engine in Panda3D and has a function of collision detection is used. The collision detection can enable the digital hand to grasp and operate an object. The Leap Motion Controller (LMC) [12] is introduced as a hand-posture sensor. A number of applications using another type of digital hand is presented in the Web site of LMC, but all of them does not have soft skin and the collision detection in real time.

The paper is organized as follows; the skeleton model of digital hands is described in Section II. In Section III, the design of soft skin is discussed. In Section IV, our idea to realize the real time operation is explained. And the demonstrations of real time operation of the digital hand are presented. In the last, the paper is concluded.

II. Skeleton Model of Digital Hand

The hand skeleton model is shown in Fig.1 based on anatomical and medical hand investigation [2].

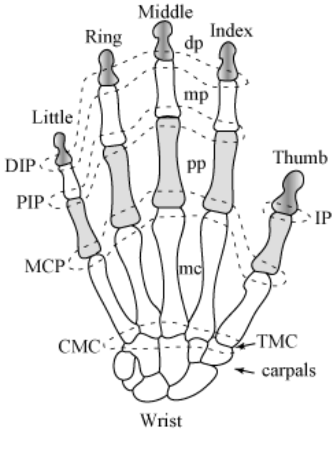


Figure 1. Hand skeleton structure.

Copyright (c) IARIA, 2016. ISBN: 978-1-61208-464-0

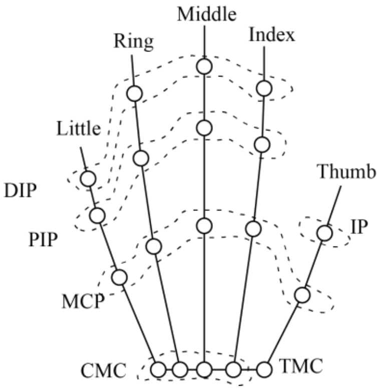


Figure 2. Skeleton model with bones and joints.

In Fig.1, abbreviated label for joints have following meanings (arranged in order from proximal to distal extremity). CMC stands for the carpometacarpal joint, MCP for the metacarpo-phlangeal joint, PIP for the proximal interphalangeal joint, and DIP for the distal interphalangeal joint. Other joint labels of thumb are: TMC for the trapeziometacarpal joint, MCP for the meta-carpophlangeal joint, and IP for the interphalangeal joint.

Each finger (not including the thumb) is composed of three bone links, called phalangeal bones. Each neighbouring pair of bone links are connected with a joint, i.e., a constraint that restricts relative translational motion of bone links in dynamics simulation. The DIP, PIP and IP has one DoF, the MCP has two DoFs, the CMC has two DoFs and the TMC has three DoFs. So, the total DoFs of human hand is 30.

It is very difficult to realize to operate a digital hand with such tremendous DoFs. Here, the joint of hand model of the LMC does not depend on DoFs, just on positions in three-dimensional Cartesian coordinate. From this, we designed that the joint of the digital hand are just connected each other. The rigid parts of the digital hand consists of the bones as rigid body and the joints, its structure is shown in Fig.2.

III. Design of Soft Skin

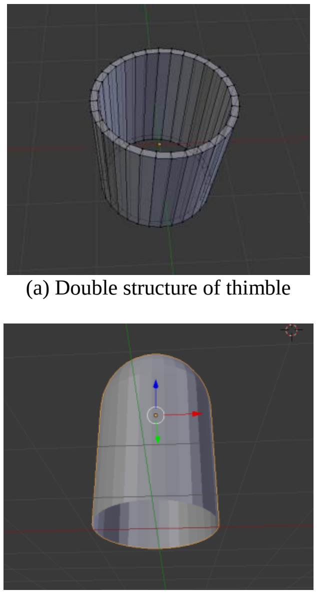
A. Design Method

There are two types of the soft skin for thimbles and fingertips, because a soft skin covering the entire digital hand makes the design and calculation load very complex. To overcome it, we designed that the soft skin covers the digital hand partially, and its design should be made to ensure the sufficient contact area between soft skin and object.

The configuration of soft skin is made by Blender, cylinder for thimbles and hemisphere with cylinder for fingertips. To give elasticity property to soft skin and convertibility to format used in Panda3D, the double structure for both is introduced. Fig.3 (a) shows a thimble which consists of two cylinders as double structure with different radius and both connected along with both edge, Fig.3 (b) shows a fingertip which consists of two couples of a hemisphere and a cylinder as double structure.

TABLE I. Parameters of Soft Skin in Bullet Physics.

kDP	Damping coefficient; damps forces acting on soft body nodes to reduce their oscillation over time. Imagine a mass hanging on a spring. Range [0,1]
kDG and kLF	Drag and Lift coefficient; relating to aerodynamics (Wikipedia_Lift,2015, NASA,2015), Range $[0,+\infty]$
kDF	Dynamic friction coefficient; just friction of nodes against surfaces, as with rigid bodies. Range [0,1]
kMT	Pose matching coefficient; be used with setPose(bool, bool). Range [0,1]
kCHR, kKHR and kSHR	Rigid, kinectic and Soft contacts hardness; controling how strict any overlap between the soft body and other types is treated. Range[0,1]


(a) Double structure of thimble

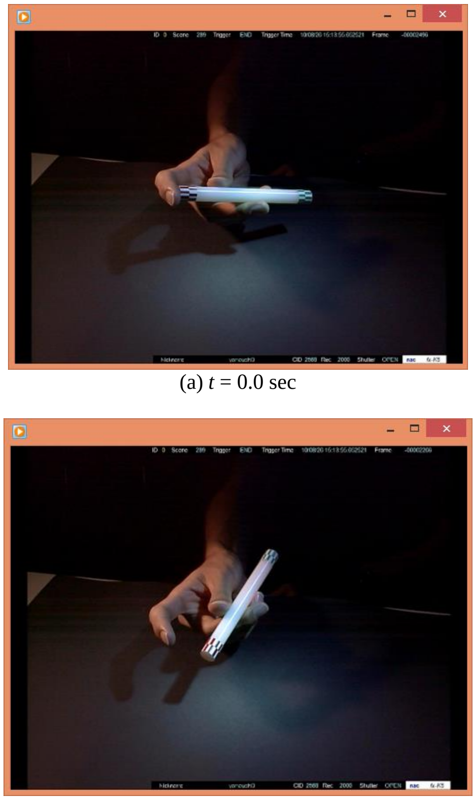
(b) Double structure of fingertip

Figure 3. Soft skin for thimbles and fingertips.

The number of mesh that makes up part of the thimbles and the fingertip will become too large, then the calculation time required for collision detection will be enormous, so it is difficult to achieve real time operation. Based on the trade-off of computational load and feasibility of the real time operation, the selection of the number is determined by trial and error.

The figure of the soft skin is introduced into soft body of Bullet Physics, and some parameters shown in TABLE I of soft body should be defined to set up it. However, the effective way to identify them have not shown yet, so we investigated that human hand played a dexterous

ISBN: 978-1-61208-464-0


ID 0 Scene 289 Trigger END Trigger Time 10/08/26 16:13:55.652521 Frame -00002496
Nickname yamauchi3 CID 2560 Rec 2000 Shutter OPEN nac 6-K3
ID 0 Scene 289 Trigger END Trigger Time 10/08/26 16:13:55.652521 Frame -00002266

(a) *t* = 0.0 sec

(b) *t* = 0.1 sec

Figure 4. Scene of dexterous operation (1000 fps)

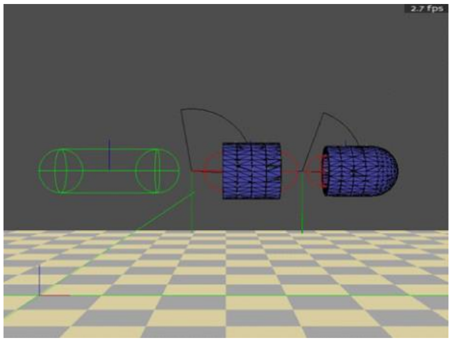
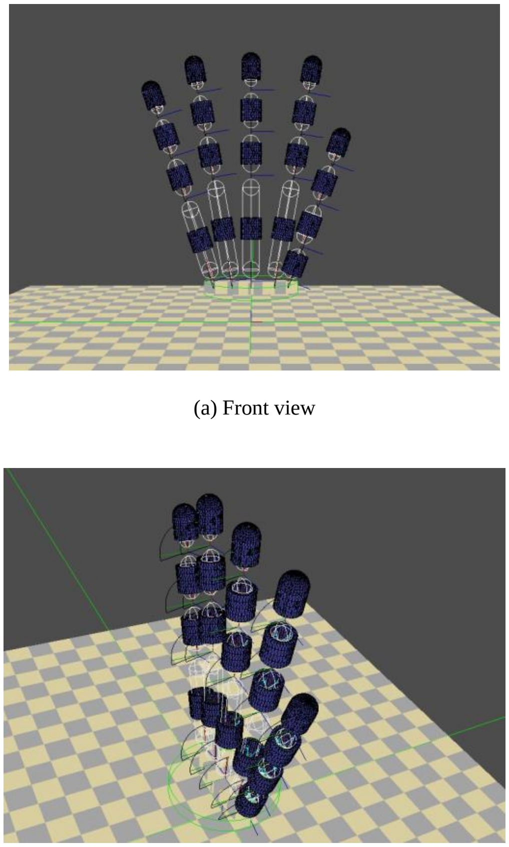


Figure 5. Digital finger with partial soft skin and rigid bone.

manipulation by using the high-speed camera (1000 fps) as shown in Fig. 4. Observing the situation of the deformable skin by investigating the movie, the parameters are adjusted to show the similar situation of the deformable soft skin.

B. Connection between Softskin and Rigid Body

A finger consists of three cylinder rigid body and the joint described in the previous section. Fig.5 shows one finger conducted by the design described above by using Panda3D. In Fig.5, the rigid bone and the partial soft skin are connected with anchors. Fig.6 shows the extending this configuration to the five fingers.



(a) Front view

(b) Overhead view

Figure 6. Digital hand, five fingers with rigid bones and partial soft skin.

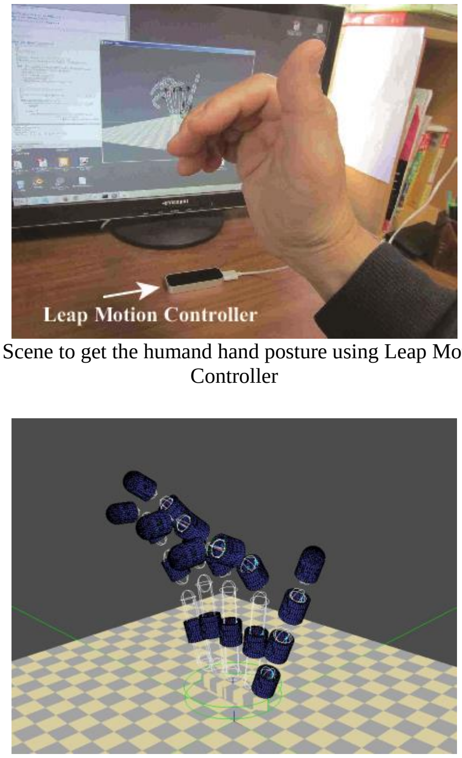
IV. Real Time Operation System

A. Hand Posture Sensing

To realize the digital hand to mimic human hand operation in real time, a sensor which is able to sense the hand posture and also the position of hand is required, then the LMC is suitable for the requirement. The LMC observes a roughly hemispherical area, to a distance of about 1 meter, and can get 3D position data of all joints of fingers and palm within sampling rate 150-295 fps (USB 3.0 connection), this is made possible by the skeleton model of hand of the LMC. Then, the position data is sent through a USB cable to the host computer.

B. Implementation

A demonstrative application has been developed to evaluate the digital hand in operation by the postures. The goal is to set up the digital hand in real time operation. The software application is executable on the CPU (Core i7-4900MQ, 2.8GHz) and the GPU (Nvidia Quadro K4100M, 1152 Cuda processors). In the system, the roles of CPU and GPU are assigned separately as following



(a) Scene to get the humand hand posture using Leap Motion Controller

(b) Digital hand chage its own posture synchronously with human hand motion.

Figure 7. Digital hand system with LMC to obtain hand posture in real time operation.

Finger Callback : CPU
Graphics Thread : GPU
Physics Simulation : GPU

These processing assigned to CPU and GPU is enable to use PyCUDA [13], because Panda3D is built in Python and the assigned has been developing in the present circumstances.

C. Experiment

The user operated the digital hand to mimic the human hand in real time processing, using the LMC as the input device of the human hand posture is shown in Fig.7. Fig.7(a) shows the scene to get the human hand posture using the LMC. Fig.7(b) shows the scene that the digital hand change its own posture synchronously with human hand motion.

We have succeeded in the real time operation. According to the movement of human figures and palm of the hand, the digital hand change its posture to mimic the hand. And when the digital hand grasp an object in the virtual physic space in which the collision detection between the digital hand is automatically calculation then the digital hand can grasp it according to the varying hand posture in real time. However, this computational load becomes tremendous, so the real time operation is not able to attain smooth execution.

V. Conclusion

This paper proposed a novel design procedure of the partial soft skin of the digital hand, and shows the connection approach with rigid bone and real time operation system. The design of soft skin and rigid body is regular way in CG creation, but the connection approach is devised because the collision detection of each body shows different phases. This approach relates on the shape of the soft skin. The reason why the partial soft skin is conducted is to reduce the computational load, but the real time operation has not shown the sufficient operation.

The real time operation is considered about the digital hand by using the LMC. The applicable demonstration in real time operation is able to be realized by tuning PyCUDA, and it will be shown in the conference stage. In the future work, the authors will strive to improve the design procedure of the soft skin and the processing time in real time operation of the digital hand.

ACKNOWLEDGMENT

This work was supported by JSPS KAKENHI Grant Number 25280125.

REFERENCES

[1] E. Y. Chao, K. N. An, W. P. Cooney and R. L. Linscheid, “Biomechanics of the Hand”, World Scientific Publishing, 1989.

[2] A. I. Kapandj, “The Physiology of the Joints Vol.1-3”, Churchill Livingstone, 2008.

[3] S. I. Edwards and D. J. Buckland, “Development and Functional Hand Grasps”, SLACK Incorporation, 2002.

[4] J. Lee and T. Kunii, “Model-Based analysis of Hand Posture”, IEEE Computer Graphics and Applications, vol.15, 1995, pp.77-86.

[5] S. Sueda, A. Kaufman and D. K. Pai, “Musculotendon Simulation for Hand Animation”, Proc. of ACM SIGGRAPH2008, vol.27, issue3, 2008, pp.1-8.

[6] Y. Endo, S. Kanai, N. Miyata, M. Kouichi, M. Mochimaru, J. Konno, M. Ogasawara and M. Shimokawa, “Optimization-Based Grasp Posture Generation Method of Digital Hand for Virtual Ergonomic Assessment”, SAE Intl J. of passenger cars-electronic and electrical systems, vol.1, issue1, 2008, pp.590-598.

[7] S. Mulatto, A. Formaglio and D. Prattichizzo, “Using Posture Synergies to Animate a Low-Dimensional Hand Avatar in Haptic Simulation”, IEEE Transactions on Haptics, vol.6, 2013, pp.106-116.

[8] H. Hashimoto, A. Sasaki, S. Yokota, Y. Ohymama and C. Ishii, “Bar Spinning as Dexterous Manipulation of Digital Hand Based on Human Hand, IASTED Intl Conf. on Modelling and Simulation, 2012, pp.413-418.

[9] H. Hashimoto, A. Sasaki, S. Yokota, K.Mitsuhashi and Y. Ohymama, “A Structure and Soft Finger Model of Digital Hand for Real Time Dexterous Manipulation”, IASTED Intl Conf. on Modelling, Identification and Control, 2014, pp.265-270.

[10] Panda3D, https://www.panda3d.org/, 2015

[11] Blener, https://www.blender.org/, 2015

[12] LeapMotion, https://www.leapmotion.com/, 2015

[13] PyCUDA, http://mathema.tician.de/software/pycuda/, 2015

Dynamic 3D Bounding Box Estimation for Video Segmentation from a Non-Stationary RGB-D Camera

Naveed Ahmed

Department of Computer Science
University of Sharjah
Sharjah, United Arab Emirates
Email: nahmed@sharjah.ac.ae

***Abstract*—We present a new method for video segmentation of RGB-D video data acquired from a non-stationary RGB-D camera. Our method uses a novel method of dynamic 3D bounding box estimation for moving objects that correctly classifies foreground and background elements of a scene even in the presence of high camera motion. Starting from the acquisition of a dynamic object using a non-stationary RGB-D camera, our method finds the mapping between different frames of the video data using the image features. The mapping between the image features derives our novel method of dynamic 3D bounding estimation that correctly segments the moving object in spite of the camera motion. The segmented video data can be employed in a number of applications, e.g., video editing, motion capture, action recognition, visual FX processing, or free-viewpoint video.**

Keywords–RGB-D Video; Non-Stationary RGB-D Camera; Video Segmentation, 3D Animation.

I. Introduction

Background segmentation is one of the important steps that is employed at the start of a number of Computer Vision algorithms. It deals with separating the foreground from the background by correctly classifying the parts of the image in either category. The algorithm when applied on the video data to separate two or more segments is called Video Segmentation. Traditionally, the video segmentation is done against a static background using the color cameras, and a number of methods have been proposed in this area [1]. The resulting segmented video from these methods is then employed in a number of applications [2] [3] [4]. A video acquired from a stationary camera is suitable for a static setting similar to a recording studio that requires some dedicated space for acquisition. The cameras are mounted at some specific points and the moving object is confined to a specific area.

In recent years, with the advent of mobile phones and smaller cameras, more and more video is captured from non-stationary cameras. In addition, the arrival of consumer grade RGB-D cameras, e.g., Microsoft Kinect [5], have opened new ways to capture true 3D video using a single camera. The captured 3D video is used in applications ranging from video games to 3D visualizations on mobile, or virtual reality devices. In order to correctly visualize a moving object captured from a non-stationary RGB-D camera, it is important to segment the video data correctly into background and foreground. The problem is very challenging because, in the data from a moving camera, the background and foreground are both moving and thus the algorithms that rely on the static background fail on this type of data.

Recently, a number of new methods are proposed in the area of segmentation of video data from non-stationary cameras [6] [7]. Lim et al. [8] estimate the fundamental matrix from frame correspondences to label the foreground and background pixels. Kwak et al. [9] extended this approach with a non-parametric framework. Zhang et al.[7] estimate the full camera motion for the true 3D reconstruction of the scene and then used the depth information to label the foreground and the background. Sheikh et al. [10] and Cui et al. [11] presented factorization-based approach from the tracked points. Narayna et al. [12] presented a method of video segmentation from a moving RGB camera using optical flow. Yi et al. [13] presented another method for video segmentation from a moving RGB camera using dual-mode Single Gaussian Model. Background segmentation using depth data has also received lots of attention in the last couple of years [14]. Koutlemanis et al. [15] presented a method of foreground detection with a moving RGB-D camera while using an initial background frame as the reference model. Zamalieva et al. [16] employed a tracklets-based method to estimate the epipolar geometry using the temporal fundamental matrix [17] of the scene and a labeling method for video segmentation.

In this paper, we present a new method for video segmentation of RGB-D video data acquired from a non-stationary camera. We start from the data acquired from a non-stationary Microsoft Kinect v2 RGB-D camera. Unlike the method from Koutlemanis et al. [15], our method does not rely on any a prior background information. Similar to Zamalieva et al. [16], we start by estimating the feature points and their matches over an interval in the video sequence. Our matching algorithm does not need dense matching, as used by [16], therefore instead of using optical flow, we only need a sparse matching. Afterward, we present a novel dynamic 3D bounding box estimation method that provides a solution in three-space, which is employed to segment all the frames of the RGB-D video sequence. The result of our work is a segmented video data using a novel algorithm from a non-stationary RGB-D camera.

In the following sections, we detail our segmentation method, first, the data acquisition and calibration is presented in Section II, afterward, the video segmentation algorithm is explained in Section III, followed by results (Section IV) and conclusion (Section V).

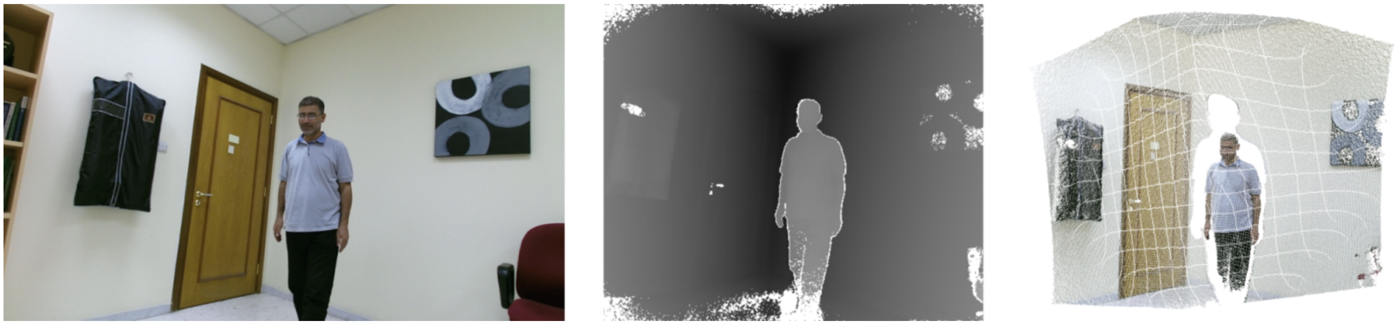


Figure 1. (left) RGB frame, (middle) Depth frame. (right) 3D point cloud resampled from mapping the depth frame to three-space coordinates. Mapping between the depth and RGB frames is used to determine the color of each 3D point.

II. Data Acquisition and Calibration

We acquire the RGB-D data of a moving subject using Microsoft Kinect v2 sensor [5]. This most recent version of Kinect can record full HD (1920x1080) RGB data at 30 frames per second. Compared to Kinect v1 (640x480), it is a significant increase. Kinect v2 can record the depth data at the resolution of 512x424 at 30 frames per seconds. In comparison to Kinect v1 depth data resolution of 320x240, the increase is modest but the underlying acquisition mechanism is enhanced to capture the higher density of depth data.

In our work, we use a single moving Kinect to acquire the RGB-D data at these default resolutions and frame rate. Our method is not confined to a single Kinect and can be easily extended to a multiple Kinect setup, as demonstrated by Ahmed et al. [18]. We capture both depth and color streams that are directly saved to the memory to avoid IO overheads and then later written to the disk once the recording is finished. Our acquisition tool is implemented using Microsoft Kinect SDK 2.0. It allows the acquisition of both RGB and depth data through a USB 3.0 interface. While acquiring the data we manually move the Kinect (rotation and translation) so that the condition for the non-stationary camera is fulfilled. To this end, we capture two images, one RGB and one depth, for each frame of the captured sequence. A captured RGB, and depth frame can be seen in Figure 1(left, and middle).

The acquired RGB and depth data have different resolutions, i.e., RGB data is stored in an 8 bit color image of size 1920x1080, whereas the depth data is stored in a 16 bit greyscale image of size 512x424. A camera acquisition setup requires a number of calibrations to determine internal (intrinsic) and external (extrinsic) camera parameters. The intrinsic camera parameters are required to determine the projection of 3D world on the camera's image plane, while the extrinsic parameters are required to estimate the camera's position in the real-world. A Kinect is comprised of two cameras, RGB and depth. Thus in addition to the intrinsic calibration of each camera, an extrinsic calibration between the two cameras is needed, so that the depth camera can be mapped to the color camera or vice versa. Moreover, the depth camera returns one depth value for each pixel in the depth image that has to be mapped to the real-world three-space coordinates, if the depth data is to be visualized in the form of a 3D point cloud. Since we are only working with a single Kinect, our method does not directly need an extrinsic calibration between multiple cameras.

We use Microsoft Kinect SDK 2.0 to determine all the required camera calibration parameters. Kinect SDK 2.0 provides the extrinsic calibration from color to depth that is stored in a 1920x1080 file with two floating point values stored for each pixel. In addition, the mapping from a depth value to the real-world three-space value is acquired after the data acquisition is finished to store the data also in the form of a 3D point cloud. A resampled 3D point cloud with the mapping to RGB data can be seen in Figure 1(right).

III. Video Segmentation

As explained in the previous section, the acquired RGB and depth data are resampled in a 3D point cloud with the RGB mapping. This 3D point cloud data shows all the acquired points comprising of the moving actor and the static background, as can be seen in Figure 1(right). In general, for most of the applications that use the video data of a moving actor for further analysis, the only relevant part of the video is the actor rather than the static background. The process of separating the foreground from the background is the well-known process of ”Background Subtraction” or in the case of video data ”Video Segmentation”, which has been employed in a number of methods for a number of years. Traditionally, most of the algorithms for video segmentation rely on the stationary camera to correctly estimate the static background [1], or rely on an initial manual estimate of the foreground or background. These methods have been proved extremely useful for a number of applications, but do not work if the data is acquired from a moving camera. In recent times, more and more data is acquired from the moving cameras, thus it is imperative to have methods that can perform the video segmentation for the non-stationary cameras.

For the non-stationary camera video segmentation, our method relies on the acquired RGB-D video data along with the mapping from depth to RGB data (Section II). Our method does not make any assumptions about the background, and does not need any initial estimate for the background. As the starting point, we extract Speeded Up Robust Features [19] (SURF) in all the RGB frames. These feature points are then matched over two frames to estimate the camera and object motion. Our main assumption is that given the moving camera, the motion of the background and the dynamic background will be different in terms of their velocity. The background motion being static will be dependent on the camera motion, whereas the foreground being dynamic will have its own motion in addition to the camera motion. It is to be noted that the background refers to the static part of the scene and it can have different depth values, and thus the depth information

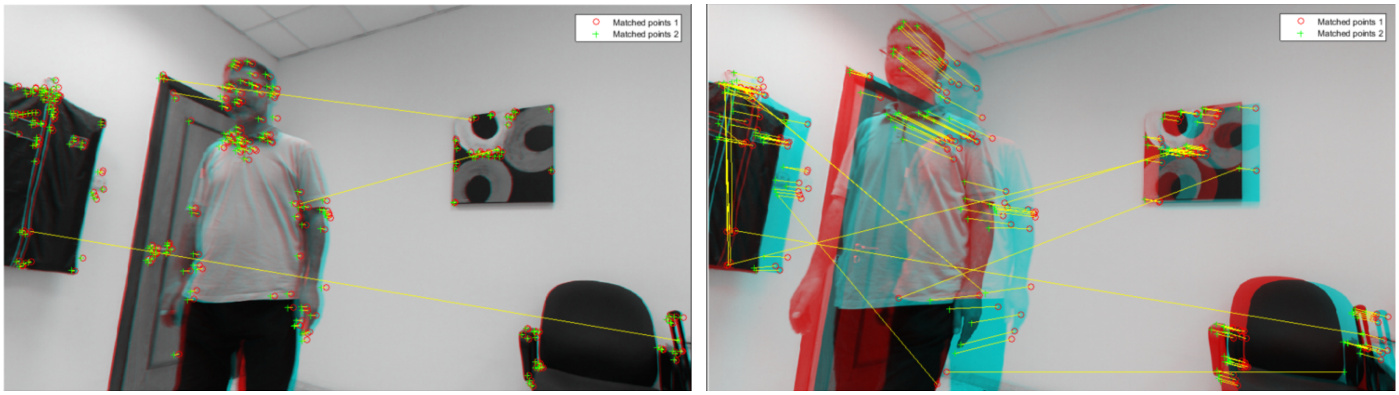


Figure 2. (left) shows the feature matching if two consecutive frames are used. (right) shows the feature matching over the interval of 10 frames. The motion is more pronounced when the frame distance is greater. Incorrect matches are automatically filtered before any additional processing.

alone cannot segment the foreground. Similarly, our method would not work if there is no movement in the foreground.

The matching of feature points provides us with the movement of each feature point over time. The matching gives both the speed and direction (velocity) of each feature point. The main assumption of our algorithm is that foreground and background should exhibit different velocity in terms of their feature points and in addition, the higher proportion of feature points will belong to the background. Kinect records data at 30 fps, thus frame to frame motion might be very small if the camera or the object are not moving very fast, and it would be impossible to differentiate between the foreground and the background based on the limited motion. Therefore, instead of comparing two consecutive frames, we compare frames at an interval of 10 frames that provides enough motion to differentiate between the potential foreground and background feature points. The interval size is chosen after analyzing the camera motion from the SURF matching. A sequence with the fast moving camera can have a smaller interval. In our method, we make sure that the matching distance should be at least greater than 2% of the image width. This satisfy the criteria of having a difference of more than 98% dissimilarity between the two frames. A comparison of feature point matching can be seen in Figure 2.

The initial match at the interval of 10 frames provides us with the starting point for our algorithm. Based on these matches, we first discard incorrect matches. The incorrect matches can occur due to the underlying SURF feature matching algorithm that is not the contribution of our work, rather we use it as it is. Incorrect matches are easier to discard using simple sanity checks based on the standard deviation of the speed and direction of all the matches. The feature matches that do not lie within the 95% of the confidence interval are discarded.

The filtered matches are then classified into multiple groups. We pick a feature match at random. The selected feature match is then used to find all the feature matches that are closer to it in terms of its velocity. We use a threshold of ±10% to find similar feature matches. Once the iterative process is finished, a group of feature matches is formed. From the remaining matches, a new feature match is randomly selected and the same iterative process of finding similar feature matches from the remaining set is repeated. This process is repeated till all the feature matches are classified in one of the groups. At the end of the process, a number of groups of matches will be found. The number of groups will depend on the type of the motion of the moving object. For example, a dynamic object with a rigid body motion couple with a static background will have small number of groups. The background will move according to the camera motion, while the dynamic object will have only one additional movement in addition to the camera motion. Other non-linear motions, e.g., a human, would depict different types of motion, i.e., arms moving in one direction while the body moves in the other direction. This combined with the camera motion will result in more groups of matches for each type of motion.

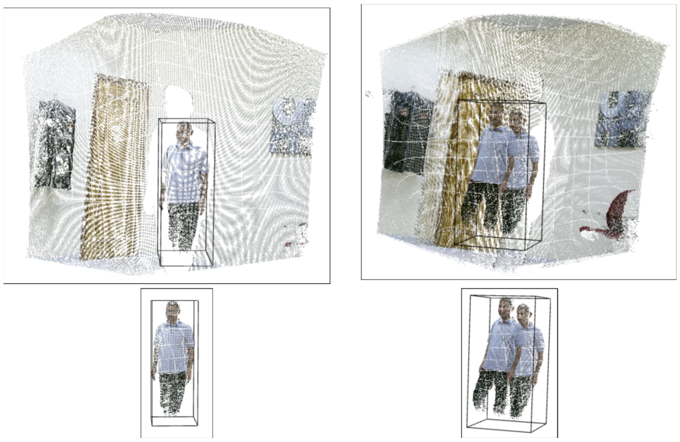


Figure 3. (left) shows the 3D bounding box localizing the foreground at one particular frame. (right) shows the dynamic bounding box spanning over the period of 10 frames that is used to segment the foreground over that interval.

In the end, once the groups are formed, our main assumption is that the group with the maximum matches will belong to the background and the rest of the groups will be classified as the foreground matches. This assumption is valid in our case, as most our scene is made up of the static background that appears to move because of the camera motion and there is only one dynamic object. If there are more than one dynamic objects in the scene then a different approach would be required to further classify the foreground into different segments, as discussed in Section IV.

Now that the feature points that belong to the foreground are identified, they are used to estimate a dynamic 3D bounding box over the interval of 10 frames. To get this bounding

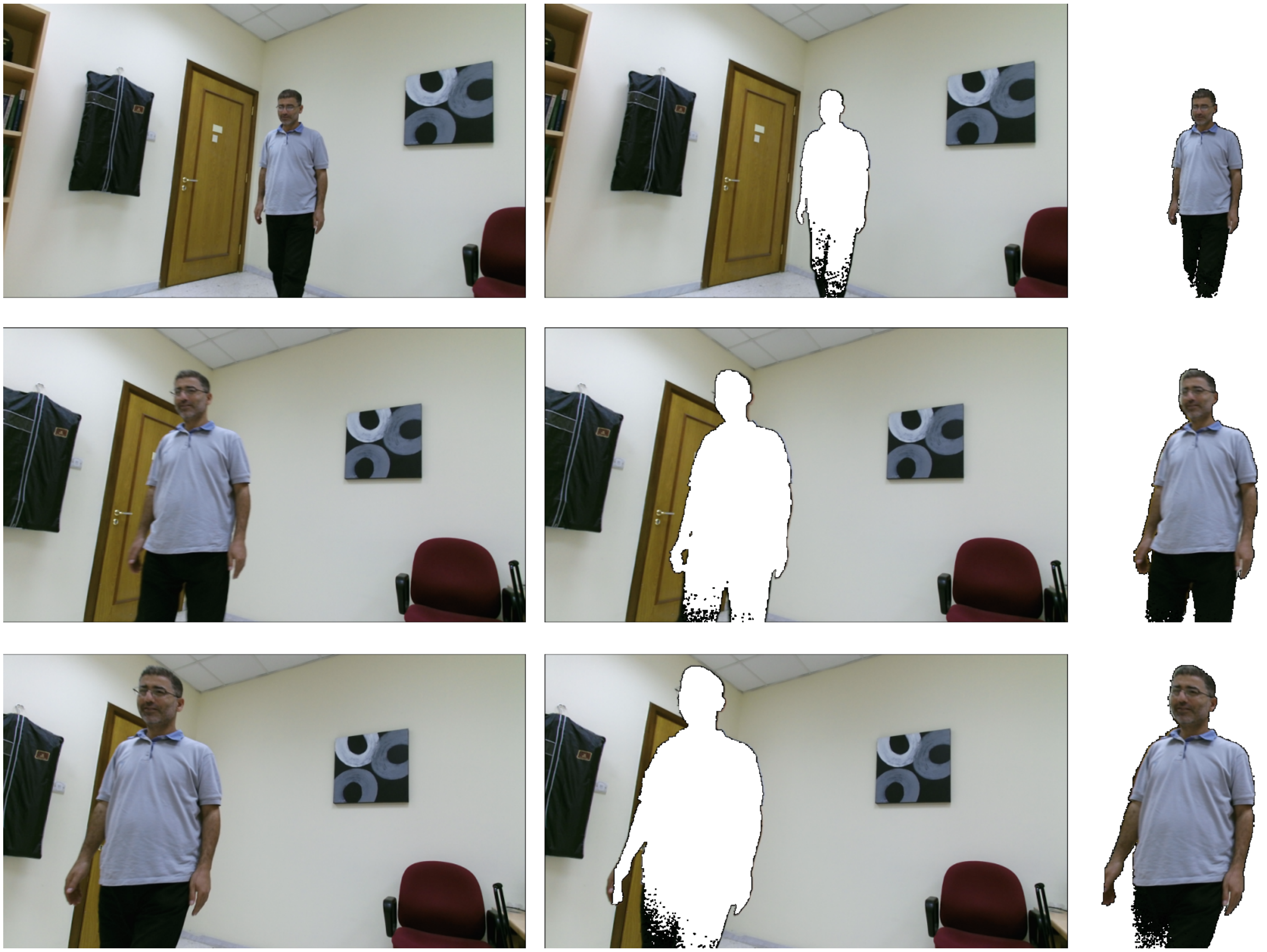


Figure 4. (left) the original RGB frame, (middle) segmented background, and (right) segmented foreground. The motion of the object and the camera is evident from these images.

box, the calibration results (Section II) are used to first find the mapping of the foreground feature points to the depth image. The depth to three-space mapping provides the 3D points corresponding to each feature point. Based on the depth data of all the feature points that belong to the foreground, we find the minimum and maximum depth value in these feature points. This provides us a 3D bounding box in the depth image at both frames. This bounding box is expanded by 20% on each side to get a conservative estimate of the true foreground. If the grown bounding box incorporates a background matching feature then its size is reduced unless the background feature point is removed from the bounding box. All the pixels in the depth image inside this bounding box at each frame are then classified as the foreground. The depth to three-space mapping also provides us with the bounding box that can segment the 3D point cloud at each frame, Figure 3(left). The minimum and maximum values of the two bounding boxes at 10 frame interval provide a stretched 3D dynamic bounding box that encompasses the motion of the object over these frames, Figure 3(right).

The stretched 3D dynamic bounding is then used to segment all the in-between frames. The approach works in both depth image, or in the three-space. All points that lie within this stretched 3D dynamic bounding box from frame 2 to 9 are classified as the foreground and the background points are discarded. The algorithm is then trivially extended iteratively over the next 10 frames till the end of the sequence. Some more results of the video segmentation can also be seen in Figure 4.

IV. Results

We recorded a data set to test and validate our method. The data set is 200 frames long and was acquired using Kinect v2 via our acquisition system (Section II). In the recorded RGB-D sequence, the object depicts a walking motion towards the camera, while the camera rotates and moves freely. Our method was able to segment the foreground from the static background completely, even with the camera motion and the classification of the foreground and background was excellent based on the visual analysis. It can be seen in the results, Figure 3 and Figure 4 that our method is able to successfully segment the foreground even in the case of high camera motion. As our method depends on the depth data for the segmentation, the slight artifacts in the results are due to the missing depth data that is a limitation of Kinect, rather than our method.

Our method is very efficient. We consider calculating SURF features and matching as a pre-processing step. In terms of actual run-time of the algorithm, it takes around a second to estimate the dynamic 3D bounding box over 10 frames and segment the remaining 8 frames. Thus, a sequence of 200 frames is processed within 20 seconds. The main bottleneck is the file output that can take additional 40 seconds for all the frames for the segmented foreground and background.

Our method is subject to a couple of limitations. The method relies on the SURF matching that depends on the quality of RGB data. In general, it works fine because Kinect v2 captures full HD video at 30 fps that results in the sharp high quality video that is suitable for finding a good number of feature points and matches under the assumption that the motion is not very fast. For a very fast motion, a higher frame-rate camera will be required. Our choice of 10 frames interval seems arbitrary but it works well in practice. The number of frames in the interval depend on the type of motion. If the motion is fast then the interval should be short and vice versa. One can quantify the interval by creating a heuristic over the velocity of feature points. If the speed is very high over 10 frames then the interval can be reduced to bring the speed within a certain limit. In future, we would like to implement this method to further automate the segmentation process. Additionally, our method works fine for a scene comprising of a single dynamic object, but at the moment we do not provide any method to further segment the foreground in case of multiple dynamic objects. It is a challenging problem that we are considering for the future work. Finally, we do not validate the goodness of our method quantitatively but rather through the visual analysis. It is to be noted that as the camera is moving there is no ground truth available for us to compare the background to the foreground. So far, we have resorted to the visual analysis of the results for the validation, but for the future work we will also generate synthetic data for the ground truth validation.

Despite the limitations, our method shows that it is possible to do segmentation of the video data from a moving RGB-D camera using both RGB and depth data.

V. Conclusions

We presented a new method to segment video data from a moving RGB-D camera. Our method uses Kinect v2 to acquire both the RGB and depth video data together with intrinsic and extrinsic parameters of both RGB and depth cameras. Initially, SURF algorithm is used to find feature points in RGB images that are matched over an interval of 10 frames to get a meaningful classification of feature points belonging to foreground and background based on their velocities. The feature points are segmented into two clusters, and the size of the clusters is used to identify the foreground feature points. The foreground feature points are then mapped to depth coordinates that are used to find a bounding box of the foreground in each frame based on the minimum and maximum depth value. The two 3D bounding boxes at 10 frames interval is then used to estimate a dynamic 3D bounding box over the 10 frames interval that is used to segment the foreground over the in-between frames. The method is then iteratively extended over the next 10 frames till the end, to segment the complete sequence. The method allows a general purpose video segmentation of dynamic objects from a non-stationary RGB-D cameras. It can be used in a number of scenarios where an RGB-D camera can be arbitrarily deployed to capture a scene without any constraints in terms of positioning the camera. In future, we plan to extend our work to include validation using the synthetic data and further segment the foreground to identify multiple dynamic objects.

References

[1] S. H. Shaikh, K. Saeed, and N. Chaki, Moving Object Detection Using Background Subtraction. Springer, 2014.

[2] J. Carranza, C. Theobalt, M. A. Magnor, and H.-P. Seidel, “Free-viewpoint video of human actors,” ACM Trans. Graph., vol. 22, no. 3, 2003, pp. 569–577.

[3] J. Starck and A. Hilton, “Surface capture for performance-based animation,” IEEE Computer Graphics and Applications, vol. 27, no. 3, 2007, pp. 21–31.

[4] E. de Aguiar et. al., “Performance capture from sparse multi-view video,” ACM Trans. Graph., vol. 27, no. 3, 2008, pp. 98:1–98:10.

[5] MICROSOFT, “Kinect for microsoft windows. http://www.kinectforwindows.org/,” November 2010, retreived: Jan, 2016.

[6] F. Liu and M. Gleicher, “Learning color and locality cues for moving object detection and segmentation,” in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2009, pp. 320–327.

[7] G. Zhang, J. Jia, W. Hua, and H. Bao, “Robust bilayer segmentation and motion/depth estimation with a handheld camera,” Pattern Analysis and Machine Intelligence, IEEE Transactions on, vol. 33, no. 3, 2011, pp. 603–617.

[8] T. Lim, B. Han, and J. H. Han, “Modeling and segmentation of floating foreground and background in videos,” Pattern Recogn., vol. 45, no. 4, Apr. 2012, pp. 1696–1706.

[9] S. Kwak, T. Lim, W. Nam, B. Han, and J. H. Han, “Generalized background subtraction based on hybrid inference by belief propagation and bayesian filtering.” in ICCV, D. N. Metaxas, L. Quan, A. Sanfeliu, and L. J. V. Gool, Eds. IEEE, 2011, pp. 2174–2181.

[10] Y. Sheikh, O. Javed, and T. Kanade, “Background subtraction for freely moving cameras,” in Computer Vision, 2009 IEEE 12th International Conference on. IEEE, 2009, pp. 1219–1225.

[11] X. Cui, J. Huang, S. Zhang, and D. N. Metaxas, “Background subtraction using low rank and group sparsity constraints,” in Computer Vision–ECCV 2012. Springer, 2012, pp. 612–625.

[12] M. Narayana, A. Hanson, and E. Learned-Miller, “Coherent motion segmentation in moving camera videos using optical flow orientations,” in Computer Vision (ICCV), 2013 IEEE International Conference on. IEEE, 2013, pp. 1577–1584.

[13] K. M. Yi, K. Yun, S. W. Kim, H. J. Chang, and J. Y. Choi, “Detection of moving objects with non-stationary cameras in 5.8ms: Bringing motion detection to your mobile device,” in IEEE Conference on Computer Vision and Pattern Recognition, CVPR Workshops 2013, Portland, OR, USA, June 23-28, 2013, 2013, pp. 27–34.

[14] K. Greff, A. Brandão, S. Krauß, D. Stricker, and E. Clua, “A comparison between background subtraction algorithms using a consumer depth camera.” in VISAPP (1), 2012, pp. 431–436.

[15] P. Koutlemanis, X. Zabulis, A. Ntelidakis, and A. A. Argyros, “Foreground detection with a moving rgbd camera.” in ISVC (1), ser. Lecture Notes in Computer Science, G. Bebis, R. Boyle, B. Parvin, D. Koracin, B. Li, F. Porikli, V. B. Zordan, J. T. Klosowski, S. Coquillart, X. Luo, M. Chen, and D. Gotz, Eds., vol. 8033. Springer, 2013, pp. 216–227.

[16] D. Zamalieva, A. Yilmaz, and J. W. Davis, “Exploiting temporal geometry for moving camera background subtraction,” in Pattern Recognition (ICPR), 2014 22nd International Conference on. IEEE, 2014, pp. 1200–1205.

[17] A. Yilmaz and M. Shah, “Matching actions in presence of camera motion,” Comput. Vis. Image Underst., vol. 104, no. 2, Nov. 2006, pp. 221–231.

[18] N. Ahmed, “A system for 360 acquisition and 3d animation reconstruction using multiple rgb-d cameras,” in Computer Animation and Social Agents (CASA), 2012, pp. 9–12.

[19] H. Bay, A. Ess, T. Tuytelaars, and L. Van Gool, “Speeded-up robust features (surf),” Comput. Vis. Image Underst., vol. 110, no. 3, Jun. 2008, pp. 346–359.

An Overview Over Content Management System Integration Approaches

An Architecture Perspective on Current Practice

Hans-Werner Sehring
Namics
Hamburg, Germany
e-mail: hans-werner.sehring@namics.com

***Abstract*—In practice, content management systems are in widespread use for the management of web sites, for intranet solutions, and for the publication of a range of documents created from diverse content. An emerging class of multimedia databases is digital asset management systems that specialize in the management of unstructured content. Despite the market for content management products aiming at integrated solutions that cover most content management aspects, there is a trend to augment content management systems with systems that offer dedicated functionality for specific content management tasks. In practice, there is particular interest in systems incorporating both a content management system and a digital asset management system. All integration forms exhibit individual strengths and weaknesses, achieved with differing implementation effort. The choice of the adequate integration architecture, therefore, depends on many factors and considerations that are discussed in this paper.**

Keywords-content management; digital asset management; software architecture; solution architecture; systems integration.

I. INTRODUCTION

Content Management Systems (CMSs) are in widespread use today for the maintenance of web sites by content producers and editors. Typical CMSs aim to manage both structured content (often in the form of hierarchies or graphs of content objects) and unstructured content, namely binary data that is shipped as some media file of a certain standard format (like, e.g., images and videos in different formats).

In practice, CMSs host elaborate processes that deal with structured content while offering only very basic functionality for unstructured content. CMS customers have an increasing demand for additional functionality for the treatment of binary multimedia content [1].

Consequently, there is a current trend to augment CMS installations with a multimedia database of the newly emerged class of *Digital Asset Management systems* (DAMs).

Both CMSs and DAMs provide a complete feature set for the management and distribution of content, the major difference being the form of content they specialize in. Since both CMSs and DAMs are designed to manage content and publish it on the web, their integration therefore is not obvious. In fact, depending on the particular requirements of a web site, different integration forms are suitable, each providing its own advantages and drawbacks.

In this paper, we discuss integration approaches for systems consisting of a CMS and a DAM. All approaches considered are derived from actual scenarios found in commercial projects. They all assume the CMS to deliver web pages and the DAM to contribute embedded multimedia documents [2]. The integration approaches differ in the point within the content lifecycle at which the DAM contributes.

The remainder of this paper is organized as follows: In Section II, we discuss the characteristics and functionality of CMSs and DAMs. In Section III, we review the lifecycle of content and digital assets, respectively, in typical CMS and DAM implementations. Section IV constitutes the main part of this paper. It presents the integration forms that correspond to certain lifecycle states. Each integration form requires some adaptations to the CMS or the DAM. These additions are discussed in Section V. Section VI presents a slight variation of the integration scenarios in the way that instead of plain assets a produced document is handed over to the CMS. The paper concludes with a summary and outlook in Section VII.

II. CONTRIBUTING SYSTEMS AND THEIR FUNCTIONALITY

With CMSs and DAMs there are two classes of systems that deal with the editing of content and shipping of content.

Both contain editing facilities including workflows and quality assurance processes. Both offer rendering and playout functionality, usually targeted at specific usage scenarios. These scenarios differ between software products (performance, editing of unique documents vs. management of uniform mass content, etc.).

As the names indicate, the systems differ in the kind of entities they deal with. CMSs focus on the management of structured content and on publication of documents that are created from compositions of pieces of content. DAMs deal with unstructured content that is managed, transformed, and published on a binary level.

Consequently, CMSs and DAMs address similar use cases, but they put a different focus on the functionalities as discussed in the subsequent subsections.

A. Content Management Systems

CMSs provide their service as follows (see also [3]).

1) Content creation: CMSs offer tools for manual creation of content by editors and for the import of content from external sources, be it from files, from feeds, or by means of content syndication.

2) Content editing: Part of a CMS is an editor tool that is used to manipulate content, to control its life cycle (see

ISBN: 978-1-61208-464-0

Section III), and to preview renderings of content. Content manipulations include adding value to content, the maintenance of description data, and the addition of layout hints and other channel-specific settings, e.g., URLs for the publication of content in the form of world wide web resources. Editing tools can be form-based with a separate preview or in-document, in which case the editor manipulates documents, and manipulations are mapped to the corresponding content. Often there are workflows to control the editing processes.

3) Quality assurance: Quality assurance for content consists of approval and publication, although in some CMS products these two activities are one. Approval marks content as being suitable for publication. Publication finally makes it available to the target audience – in the form of rendered documents. Quality assurance should be embedded in the CMSs workflows.

4) Rendering: Rendering is the process of creating documents from content. Structured content typically is rendered by mapping content structures to document layouts. The ability to manipulate binary content is limited compared to that of a DAM with matching capabilities. CMSs offer general functionality on media content suited for a particular publication channel, e.g., for the web. This particular case includes rendering of images for adaptive design, e.g., to resize them for specific channels or to apply device-specific format conversions.

5) Playout: The shipping of rendered documents, called delivery or playout, is not necessarily a core functionality of a CMS. But since playout usually is tightly coupled with rendering, CMS products include a playout component. Some CMSs target high performance output, sometimes being integrated with Content Delivery Networks (CDNs).

B. Digital Asset Management Systems

A DAM's functionality includes the following [4].

1) Asset Creation: Assets are created in a DAM as content is in a CMS, manually or in automated processes. Manual creation is typically accomplished by means of an external authoring tool. Its output is uploaded to the DAM.

2) Asset Editing: Editing is typically restricted to the maintenance of structured information (descriptive data, e.g., defining time code information in moving image, legal information, provenance information, etc. [5]). Binary manipulations are performed by authoring tools. Editing may take place in workflows [6].

3) Quality assurance: DAMs have an approval process like the one of CMSs. Workflows for quality assurance can typically be customized.

4) Rendering: The rendering of digital assets consists of format conversions, media manipulations, and generating multimedia documents from multiple assets. Transcoding particular video formats for different browsers or mobile platforms is a typical manipulation task. Manipulations include image manipulation, e.g., scaling of images for adaptive design, inserting logos in photos, watermarking of documents, etc. An example for on-the-fly document generation is assembling a video from moving image and sound for multilanguage videos. Whole hypermedia documents can theoretically be created this way. Another example is the addition of descriptive data to multimedia assets as meta data, e.g., Exif data.

5) Playout: DAMs typically can deliver assets, at least by shipping online to the web or offline by creating files, e.g., for print. Some DAMs offer more sophisticated playout functionality, e.g., reliable delivery, at-most-once delivery, exactly-once-delivery, or digital rights management. DAMs specialized in video management offer a playout based on QoS parameters. In particular, they measure network latency during video transmission to be able to sacrifice image quality in favor of synchronicity if needed [7].

III. CONTENT AND DIGITAL ASSET LIFECYCLES

Both content objects managed by a CMS and assets managed by a DAM have a lifecycle. In most products, these lifecycles are explicitly represented by states of the objects. Figure 1 illustrates the states and possible state changes as described below.

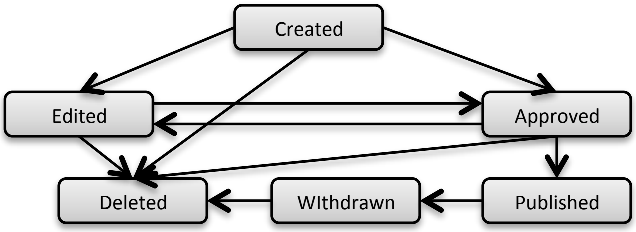


Figure 1. Lifecycle states of content objects.

The content object lifecycle starts with content objects being created. This can happen manually or by importing external content, e.g., from files or news feeds.

Subsequent editing adds value to content. Changes affect the actual content or descriptive information that is also stored in content objects. In particular, editing may include linking content objects to each other in order to create multimedia documents from the resulting object graphs.

Quality assurance for content is reflected in a dedicated approval step that marks content as being suitable for publication. Such content is, depending on the CMS product, either directly available for rendering and shipping or it constitutes a candidate for a final publication step. In the course of this paper we draw no distinction between publication and approval.

An approved object that is edited becomes unapproved. Typically CMSs support versioning of content and this way allow the approved version to be online and a newer version to be edited.

In many states, a content object can be deleted.

Assets, being a different form of content, have a similar lifecycle. They are initially created inside a DAM, be it by import from external sources or by original authoring and

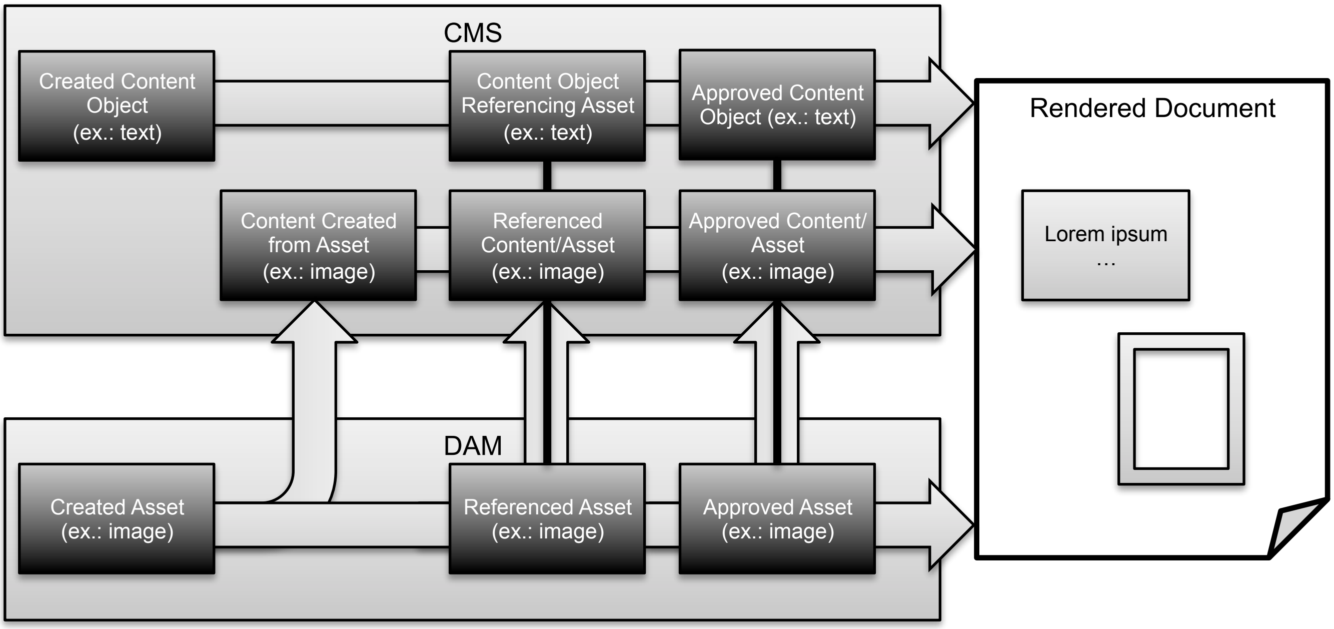

CMS
Created Content Object (ex.: text)
Content Object Referencing Asset (ex.: text)
Approved Content Object (ex.: text)
Content Created from Asset (ex.: image)
Referenced Content/Asset (ex.: image)
Approved Content/ Asset (ex.: image)
DAM
Created Asset (ex.: image)
Referenced Asset (ex.: image)
Approved Asset (ex.: image)
Rendered Document
Lorem ipsum
…

Figure 2. Example content and asset lifecycle and relationships.

storing the results inside the DAM. Editing assets is not a primary use case of a DAM [8], so we omit asset modifications here. DAMs support quality assurance by an approval process, though, similar to that found in CMSs.

IV. Time of Asset Integration

Even if the management of structured and that of unstructured content are separated utilizing a CMS and a DAM, respectively, content and assets need to be combined in published documents.

There are various integration scenarios to achieve this kind of separation of concerns. For a concrete system the integration approach should be chosen based on the requirements that the system needs to fulfill and the implementation effort. On that basis, the most beneficial approach can be chosen.

Each integration form has its specific advantages and disadvantages and addresses a different set of requirements. The subsections of this section discuss one approach each.

The subsequent Section V discusses the implementation effort of each integrated solution.

For the integration scenarios we only consider the case of a CMS being used to prepare content and to define how to render documents. This is the particular strength of a CMS that cannot be substituted by a DAM. Therefore, the CMS will always be in lead when considering the overall document publication process.

The approaches thus differ in the point in time at which an asset is integrated into the CMS. **Figure 2** illustrates the scenarios covered in this paper by different content flows.

A. Integrating Assets at Playout Time

The integration at playout time makes full use of the DAM's functionality with respect to rendering and playout. Documents are created from both content and assets at the latest point in time possible. This way, it is the loosest integration form that happens at the point of document assembly. The equivalent in an information system is the presentation layer.

Though this frontend integration makes this approach the most volatile one, it is often preferred in practice due to its comparably low implementation costs and due to the fact that all of the DAM's functionality is being used.

A CMS's editor tool allows content objects to be related to each other. Such relationships are required either to be able to link documents or to define content structures that lead to documents composed of various content objects. Figure 2 uses the example of an image related to text. This integration scenario – as well as all the other ones discussed in the course of this paper except for the integration at creation time – requires an extension of the CMS's editor tool with a search in the accompanying DAM. At the same time the search functionality of the DAM is required to be exposed to the CMS.

For integration at playout time the CMS stores proxy content (as asset references) only at editing time. Such proxy content represents an asset from the DAM. It is created when an asset reference is defined using the editor tool.

The external references from proxy content to the asset it represents require the DAM to provide stable external asset IDs or addresses.

The CMS renders proxy content objects as references to the according assets residing inside the DAM that delivers them directly into the documents.

After creation of proxy content the CMS needs to receive events concerning the asset's lifecycle. A referenced asset might become unavailable for publication later on due to disapproval or deletion from the DAM.

There is no general way to prevent possible runtime errors due to assets that have been deleted or ones that have otherwise become inaccessible. Depending on publication strategies all content referencing such an asset may become inaccessible, as well as (transitively) all content referring to such content. In other cases, it might be possible to remove such references but leave the rest of the content intact.

In the case of web content management this scenario requires the DAM to be exposed to the Internet in order to be able to deliver the assets for inclusion into documents.

B. Integrating Assets at Render Time

Like most of the integration scenarios this one requires (see previous subsection): an extension of the CMS's editor with a search in the accompanying DAM, capabilities to manage asset references in order to relate assets to content, and means to deal with the fact that asset and content life cycles cannot be synchronized in a generic way.

During rendering, references are resolved. Assets are transferred to the CMS and stored at least in the public stage. The benefit of this step is increased independence from the asset lifecycle from this point on: asset deletion no longer leads to inconsistent publications out of the CMS. Nevertheless, disapproval of an asset does not automatically lead to withdrawal of corresponding and referring content.

The problem with unavailable assets exists as in the preceding case. Yet it does not occur at playout time, but instead at rendering time. This makes no difference in most contemporary CMSs. In offline CMSs that render documents in advance, this can be beneficial, though.

C. Integration Assets at Approval / Publication Time

This integration scenario is much like the preceding ones, only that it integrates assets even earlier in the asset/content lifecycle, namely during approval or publishing.

Typically, content is published in a transitive way. E.g., when an article is published, all related images need to be published in the same step as well, or otherwise the publication of the article will fail.

This integration scenario is based on an extension of the CMS's approval process in a way that assets are retrieved from the DAM and stored as content in the CMS during the process (based on proxies created at editing time), at least in the public stage. This scenario is based on the assumption that it is insufficient to apply quality assurance to the proxies alone because of asynchronous asset modifications in the DAM. Instead, the assets' approval state is checked as part of the approval process of the CMS.

In contrast to the preceding scenarios, the CMS is leveraged from having to consider unavailable assets at playout time. Still, the decoupled life cycles of asset and corresponding content need to be dealt with. To this end, there either needs to be a synchronization of asset and content state based on notifications as discussed before, or the CMS neglects the approval state in the DAM and maintains the state on the basis of content objects only.

In this integration scenario, as opposed to the preceding ones, the CMS's publication, rendering, and playout capabilities are used for digital assets. Section V.B discusses the resulting implications. The DAM's playout functionality (see Section II.B) will not be utilized.

D. Integration Assets at Editing Time

Assets can be added to the CMS at editing time, e.g., when a reference to an asset is added to some content. This requires an extension of the CMS's editor with (a) search in the accompanying DAM like in the cases above and (b) on-the-fly content creation from selected assets.

If assets are integrated in the CMS before approval they need to be monitored for subsequent changes. To this end, there needs to be synchronization once content has been created from an asset. This synchronization may be eager (on every asset change) or lazy (on demand, e.g., at playout time).

With integration at approval time and before, rendering and playout are performed by the CMS (s.a.).

E. Integrating Assets at Asset Creation Time

The earliest possible integration of assets is at the time of their creation: assets are added to the CMS as soon as they are created in the DAM.

This scenario only makes sense if the DAM is also used in processes other than document production through a CMS. Otherwise there would be no need for a DAM at all. When assets still have an independent lifecycle inside the DAM then the integration requires continuous synchronization. This synchronization is performed eagerly in order to provide assets as content for selection within a CMS. There is no need for an extended editor that allows searching the DAM since assets can directly be found in the content base.

In this scenario, nearly all DAM functionality is neglected in favor of the corresponding CMS functions. As in the above scenarios quality assurance is controlled by the CMS, and rendering and playout are carried out solely by it.

V. Required System Adaptations

In order to implement the integration of a CMS with a DAM in one of the forms presented in the preceding section, some extensions or adaptations to the software products are required. Table I gives an overview of required adaptations and attributes them to the integration scenarios.

A. Added Functionality

The scenarios that rely on a continuous synchronization of assets and corresponding content objects are typically implemented through notifications by events, e.g., the event of an asset having been modified. In these scenarios the DAM needs to be an event source and the CMS an event subscriber. The DAM will produce events and transmit them to subscribers. The CMS registers for such events and to interpret them. When this functionality is not found in the CMS (which usually is the case), there needs to be an external software component that listens to such events and then triggers some actions inside the CMS. To this end the CMS needs to provide an externally usable API.

In order to relate events to content created from assets, the DAM has to provide stable IDs or addresses (like, e.g., URLs) of assets. This is particularly important due to the fact that assets are long-lived.

Most events are related to specific revisions of assets. For those events subscribers need IDs that reference asset

TABLE I. CHANGES TO SOFTWARE PRODUCTS DEPENDING ON ASSET INTEGRATION TIME

Aspects	**Form of Integration**				
	Creation time	***Editing time***	***Approval time***	***Render time***	***Never***
Changes to CMS	• subscribe to and listen to events (from DAM) or expose public API; create content on asset creation or modification	• media selection dialog changed to query DAM • on-the-fly content creation upon asset utilization (linking) • subscribe to and listen to events (from DAM) or expose public API; modify content on asset modification	• media selection dialog changed to query DAM • surrogate objects for assets • on-the-fly content creation on public stage upon asset (proxy) approval • check of asset's approval state upon asset proxy approval	• media selection dialog changed to query DAM • surrogate objects for assets • on-the-fly content creation on public stage upon asset (proxy) rendering	• media selection dialog changed to query DAM • surrogate objects for assets
Changes to DAM	• event source for CMS • stable external IDs (to relate assets in events)	• query interface for CMS • event source for CMS • stable external IDs (to relate assets in events)	• stable IDs/addresses • query interface for CMS • interface to query approval state from CMS	• stable IDs/addresses • query interface for CMS • event source for CMS	• stable IDs/addresses • query interface for CMS
Unused CMS functionality			• quality assurance	• quality assurance • rendering (assets)	• quality assurance • rendering (assets) • playout (assets)
Unused DAM functionality	• rendering • playout	• rendering • playout	• rendering • playout	• playout	

revisions, not assets in general. For an example of IDs fulfilling this requirement see the CMIS object IDs [9].

As described in the preceding section, some integration scenarios rely on an asset selection dialog integrated into the CMS's editing tool. Usually, such a dialog exists, but is used to select multimedia content from the CMS itself. This dialog has to be extended in a way that allows picking assets from the DAM that have not previously been imported into the CMS. Such a dialog must furthermore be backed by functionality to create content from the chosen asset, either with a copy of the content or with a link to the asset. In order for the asset selection to work the DAM has to offer search functionality to the CMS (editor). The search result contains, depending on the scenario, the asset data or the asset ID or address.

B. *Unused Functionality of the Software Products*

There exists functionality that is provided both by a CMS and a DAM. In an integrated system the corresponding functions of one the systems may not be used. From an architectural point of view, this makes no change. But certain strengths and weaknesses of the products might not be considered in an optimal way in particular integration scenarios.

In those integration scenarios where the CMS handles references to assets in the DAM only, the quality assurance measures, usually some approval process, of the CMS are not in effect for assets. Approving a content object just makes a statement about a version of the corresponding asset at approval time, but assets may change without the handles inside the CMS being altered.

The aforementioned event-based synchronization can be used to monitor the approval state of assets and to adjust the approval state of the corresponding content objects. But considering the whole asset lifecycle there are situations that cannot be handled. The most drastic example is a valid asset that is (rightfully) referenced by published content. If now the asset is deleted then the CMS notices the state change. But it cannot decide whether to keep the image reference (thus rendering documents with missing images), whether to remove the images reference from all content objects (thus automatically altering the content; an operation that is usually unwanted in CMSs), or whether to disapprove all content objects containing the image reference (an operation that has to be applied recursively and can thus have unexpected effects).

If integration of a CMS and a DAM takes place in a way that assets are copied to the CMS before playout time, the rendering and possibly playout functionality of the DAM will not be utilized. This is a major drawback of those integration scenarios since these are about the most powerful contributions of a DAM. A CMS typically offers very limited rendering functionality for multimedia content, if any (see Section II.A). In the subsequent Section VI, we discuss an integration approach that allows to use more of a DAM's rendering functionality. Playout with QoS parameters is usually not provided by a CMS, but by some DAMs.

If integration of a CMS and a DAM takes place at a point in the asset lifecycle later than content editing, the rendering and possibly playout functionality of the CMS is not used for content originating from assets. As pointed out above, the corresponding functions of a DAM are typically more powerful that those of the CMS (see Section II.B). But there are some things to consider in specific scenarios.

The rendering of assets often is influenced by context-specific parameters of the publication channel at hand. For adaptive web design, for example, images are scaled to the actual screen size of the device posing a request, videos are transcoded to suitable formats, etc. In addition, some CMS installations allow editors to define the image formats used in particular situations, e.g., renderings in certain contexts. This cannot be achieved as easily when the DAM has the duty of rendering assets.

With respect to playout a CMS does not provide the media-specific functionality found in a DAM, in particular there is no quality-controlled adaptive playout. On the other hand, the CMS uses a playout infrastructure consisting of sophisticated caching, inclusion of content delivery networks, etc. This infrastructure has partly to be made available to them DAM.

VI. Advanced Scenario: Asset Shipping to CMS

From an editing viewpoint the integration at the time of asset creation or editing time is the most beneficial. To allow more of a DAM's rendering functionality to come into play in such an integration scenario, a variation of the corresponding integration approach can be taken.

In the preceding section we assumed the systems to pass "raw" content to the other, limiting the DAM to a multimedia database. Alternatively the synchronization of asset content can be considered a logical playout step from the DAM with the CMS being the receiver of rendered documents.

Though this variant does not help for playout (QoS parameters, etc.), it allows the integrated system participating in the DAM's functionality to render multimedia content (see Section II.B).

Particular attention has to be put on the interplay of the DAM's and the CMS's media manipulation functionality. E.g., a graphic would be stored in raw format inside the DAM. It provides a rendered version to the CMS, e.g., in a predefined format and resolution. During the shipping of the content from within the CMS this will in turn prepare the graphics data by scaling it for the usage at hand (full screen version, smaller embedded version, high resolution print version). The concatenation of the manipulation functions may lead to quality losses compared with a one-step rendering through the DAM's rendering functions.

In cases where there is no interference between the DAM's and the CMS's rendering of assets, the concatenation allows combining the quality of renditions provided by a DAM and the control over renditions by a CMS editor.

VII. Summary and Outlook

The paper closes with a summary and an outlook.

A. Summary

This paper presents various forms of integration of a CMS and a DAM. If the CMS is in lead regarding the overall content management process then the main difference between the integration forms is the point in the asset lifecycle at which an asset is introduced in the CMS.

All integration forms exhibit individual strengths and weaknesses, achieved with differing implementation effort. The choice of a suitable integration form, therefore, depends on many factors and considerations discussed in this paper.

B. Outlook

For integrated solutions – like a CMS combined with a DAM in this case – we would like to see a repository of typical requirement/solution patterns.

The discussion in this paper shows that many decisions rely on the particular properties of the software products used. The solution scenarios should therefore be refined to consider actual software products with their individual capabilities to be of increased value in practical applications.

Furthermore, some decisions have to be made on the basis of more concrete requirements: the integration approach in general, but also implementation details like, e.g., the way how to handle concurrent asset modifications in the DAM and in the CMS. A comprehensive catalog containing more refined use cases and blueprints for typical solutions is required in practice.

Future work will try to extend the considerations to more general integration scenarios in the field. A quite prominent example is product information management fulfilled by, e.g., a CMS in cooperation with catalog management or a CMS combined with a shop solution.

Acknowledgment

The author thanks his employer, Namics, for the opportunity to follow his scientific ambitions by publishing some of his thoughts. In particular since these relate to and extend commercial activities.

Fruitful discussions with numerous colleagues (current and former), business partners, and customers led to the insights presented in this paper. My thanks go to all of them.

My thanks also go to the anonymous reviewers for the constructive hints that helped to improve this paper.

References

[1] Ovum, Making the case for digital asset management in retail: Using technology to manage digital assets effectively. Whitepaper, August 2015.

[2] A. Saarkar, Digital Asset Management. Whitepaper, Cognizant Technology Solutions, 2001.

[3] S. King, "Web content management", in Computer Technology Review. Los Angeles, vol. 22, issue 11, p. 9, 2002.

[4] D. Austerberry, Digital Asset Management: How to Realise the Value of Video and Image Libraries. Amsterdam, Boston: Focal Press, an imprint of Elsevier Ltd., 2004.

[5] Y.-M. Kim et al., "Enterprise Digital Asset Management System Pilot: Lessons Learned", in Information Technology and Libraries, John Webb, Ed. vol. 26, no. 4, 2007.

[6] T. Blanke, "Digital Asset Ecosystems: Rethinking crowds and cloud", Chandos Publishing, 2014.

[7] H. Thimm and W. Klas, "Playout Management in Multimedia Database Systems", in Multimedia Database Systems, K. C. Nwosu, B. Thuraisingham, and P. B. Berra, Eds. Springer US, pp. 318-376, 1996.

[8] C. D. Humphrey, T. T. Tollefson, and J. D. Kriet, "Digital Asset Management", in Facial Plastic Surgery Clinics of North America, vol. 18, no. 2, pp. 335-340, 2010.

[9] Content Management Interoperability Services (CMIS) Version 1.1. 23 May 2013. OASIS Standard. [online]. Available from: http://docs.oasis-open.org/cmis/CMIS/v1.1/os/CMIS-v1.1-os.html

Melody Transcription Framework using Score Information for Noh Singing

Katunobu Itou*†, Rafael Caro Repetto†, Xavier Serra†
* Faculty of Computer and Information Sciences, Hosei University, Tokyo, Email: it@fw.ipsj.or.jp
† Music Technology Group, Universistat Pompeu Fabra, Barcelona

***Abstract*—Not only do novice listeners have difficulty enjoying Noh singing but researchers also have difficulty treating it formally. A major reason is the huge difference between the score information and the acoustics of its execution. This paper proposes melody transcription for Noh singing using score information. The method's design is based on comparative observation among score information, commentary, and the acoustic signals of its execution by multiple performers. The calculation is based on global polynomial regression and modification within the note using score information. According to visual judgment of the plots, the resultant transcriptions fitted the pitch contours well. In addition, the possibility of discovering new unprescribed ornaments using melody transcription was suggested.**

Keywords–Noh singing, Melody transcription, Speaker adaptation, Phone segmentation, Music

I. Introduction

Noh is a traditional Japanese performance art consisting of music, drama, and dance.

In Japan, one can see a Noh performance nearly every day, with many audience members being repeaters. For a first-timer, watching the dance and listening to the music is comprehensible, however understanding the drama and the songs is difficult. Indeed performers' phonation differs from that of modern Japanese. As for the musical aspect, even the principle of the scale is completely different from that of modern music. For these reasons, enjoying Noh requires a certain amount of experience and familiarity.

Moreover, Noh singing seldom becomes a subject of research. One main reason is the greater difference between the acoustic signal of singing and score information, because the pitches of scale notes are not absolute and are changeable even within a single phrase. In previous Noh music research, several studies were concerned with the acoustics of the singing voice [1]–[3]. For the melody of Noh singing, previous research has only dealt with the interpretation of score information contained in vocal books [4], [5]. In the research about the acoustics of melody, there was only a comparison study [6].

Therefore, we propose a framework of melody transcription for Noh singing using score information to help to bridge the gap between the acoustic signal and the score. Visualization of the resultant transcription may help novice listeners to interpret the musical aspect of Noh singing, and researchers may use it to ascertain the melodic line. In addition, without such a framework, importing the recent improvement of computer music research, for example, music information retrieval, and synthesis is difficult.

In section II, the musical aspects of Noh singing are described. In section III, the available information to be utilized for transcribing the melody is described. In section IV, the proposed melody transcription method is described. In section V, the results of the preliminary evaluation are presented, followed by a discussion. Finally in section VI, we conclude the paper.

II. Musical aspects of Noh Singing

Noh music consists of vocal and instrumental parts. For the vocal part, the main actor (*shite*), the second actor(*waki*) and a few subsidiary actors sing the main melody and the chorus is sung by the accompaniment singers(*ji*).

The melody of Noh singing has three modes: the melodic mode (*yowagin*), the dynamic mode (*tsuyogin*), and the speech mode (*kotoba*). The lyrics (also known as script or words) are written in classic Japanese.

The melodic and dynamic modes have their own scale. Both scales have two or three main notes, and each main note has its auxiliary notes. The skeleton of the melody is composed of the main notes, and the next notes are strictly limited by composition rules. In addition, in a phrase, many notes stay at the same pitch. Consequently, the variation of the types of the melodic line on the score is more limited than that of other music.

In contrast, in the execution of the score prescribed melodic line, the scale is not absolute. Even within a single phrase, the same pitch notes under the score information can be sung with a different actual pitch (f0, fundamental frequency), and the difference between different pitch notes is not absolute either. An actual pitch varies depending on singers and varies even within a single piece depending on singing styles. Actual pitch also varies depending on the characters in the drama.

In Noh singing, ornaments are huge and heavily used. Ornaments are categorized into two types: one is prescribed in the score and the other is not prescribed. Examples of prescribed ornaments are *"hon-yuri"*, a type of melisma at the end of phrase, and abrupt rising and falling pitch (float). Examples of unprescribed ornaments are vibrato, sliding from a lower pitch at the beginning of a phrase, and stress at the end of a phrase.

For these reasons, in interpreting Noh music, an expression and/or personality included in actual execution of the melody is more important than the melodic line prescribed in the score. Such an expression and/or personality deviates from the exact execution prescribed by the score. According to these characteristics, for the Noh melody, there is no standard for measuring the accuracy of performances in acoustics. In this study, we propose melody transcription aimed to be utilized as a standard of the melodic line to measure expression or individuality. In order to transcribe from acoustic signals, score information along with the knowledge of interpretation is used.

III. Information for transcribing

There are five *shite* schools and they publish their own vocal books(*utai bon*), which contain all the lyrics of a piece, and the parts of the melodic and dynamic modes have melodic annotations. These annotations prescribe the melodic line similar to scores in Western music. However, some information is implicit and loose, so these annotations have to be interpreted as context dependent or based on common knowledge within Noh music. Vocal books are used for amateur performers' practice; however, they found melodic interpretation difficult, so commentaries were also published [7].

In the commentary, to assist interpretation, graphical notation of the melodic line is used. However, this notation does not assist understanding of the melodic line in the way it does in Western musical scores, because of the difficulty in estimating the exact line from acoustic signals. The commentary includes the graphical notation of 300 phrases from 55 pieces. Figure 1 shows an example of a graphical notation [7] and Figure 2 shows a pitch (f0) contour of its execution. This sample is a phrase in the dynamic mode.

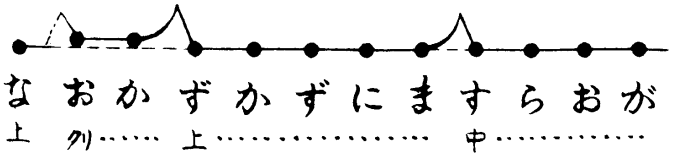


Figure 1. A graphical notation of a melodic information in a Noh vocal book

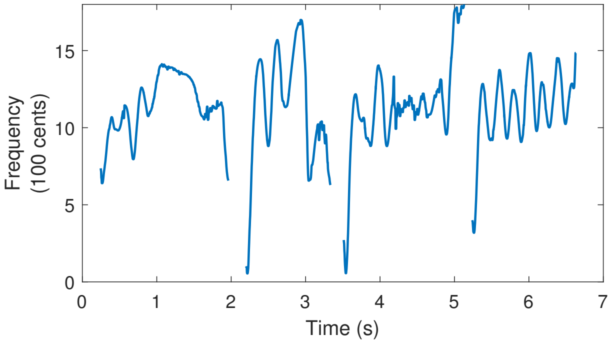


Figure 2. Pitch contours of Noh singing

In Figure 1, the first row is the graphical notation. Bullets show onsets, and lines and curves show pitch transience. In this notation, the different vertical positions indicate different pitches similar to Western music notation. However, this notation is continuous unlike the discrete Western notation. Therefore, ornaments indicated using smaller symbols, e.g, a grace note, are indicated using the combination of curves. The second row contains the lyrics in Japanese *kana*. The last row contains note names. Figure 2 illustrates the difficulty of finding onsets and transitions of notes because of the extent of certain vibrato being greater than the transition pitch difference. By contrast, the extent of vibratos in Western music hardly exceeds 200 cents. Cent is a unit for musical pitch. One semitone is 100 cents. In this paper, n cents for an fHz pitch is $n = 1200 \cdot \log_2 \frac{f}{131}$. As seen in this example,fitting a melodic line to a pitch contour is difficult because a new melodic line notation, closer to the acoustic signal, is required.

To design the new notation, we collected speech signals of phrases in the commentary from a cappella parts in commercial Noh singing compact discs (CDs) and observed their spectrograms and pitch contours. Consequently, we categorized the pitch transition patterns within a single note into the following three: maintaining same pitch (staying), transition to a different pitch, and abrupt rising and falling pitch (floating). In every pitch transition, many ornaments–vibratos, pitch sliding, and others–were observed.

In the next section, we describe the proposed melody transcription method on the basis of the collection and the categorization above. This method uses score information, and its input is assumed to be an a cappella signal.

IV. Noh Melody Transcription using Score Information

A. Input signal

The proposed method assumes a single phrase as an input. Normally, Noh-singing CDs are edited as a whole piece that runs from 10 to 40 min as several tracks or a single track. However, in vocal books, phrases are divided with punctuations, which correspond to rest marks of music. In Noh-singing signals, these punctuations almost correspond to pauses, and such pauses are automatically detected when using a voice activity detection technique of speech recognition.

B. Score information

Each note corresponds to a single syllable. In Japanese, syllable is classified in two types: CV, which means a consonant succeeding a vowel, V, which means just a vowel. Hence, each note has one or two phone fields. In addition, each phone field has pitch transition information, which consists of three values at most. The first is an original and mandatory pitch. The second is a first-transited pitch, and the third is a second-transited pitch. Pitch is expressed as an integer whose value difference refers approximately to a halftone of Western music. Table I is an example of the first three notes in Figure 1.

TABLE I. Score information

syllable	*phone*	*pitch*		
na	n	5		
	a	5	10	6
o	o	6		
ka	k	6		
	a	6	10	5

In Western music, pitch transition occurs at the onset. In Noh singing, pitch transition does not occur quickly at the onset, and the execution is like a grace note [1]. In addition, unlike Western music or Japanese popular music, consonants last longer. Reflecting these aspects, a note must be divided into phones, i.e., a note must contain a phone boundary.

C. Pitch contour estimation

A pitch contour is estimated as an f0 contour using Melodia algorithm [8]. To be adjusted to Noh singing, the following condition is changed. Noh singing has a very low pitch, whose lower values can be less than 100 Hz; hence, we do not apply the lower part of the equal loudness filter [9]. In addition, we do not use voicing detection because the voice quality of Noh singing differs from that of Western music singing [2] and musical instruments.

D. Score alignment

Score alignment is a technique to link score information to audio signals of a score's performance. For a monophonic source, in order to link, the detected onset of note [10] or the shape of pitch contour [11] is used. However, these techniques are not suitable for Noh singing. Many onsets of Noh singing are blurred because each syllable is not uttered separately. As mentioned above, the melodic line of Noh singing often stays at the same pitch, and such a situation, of course, does not change pitch contour shapes. In addition, vibrato depth in Noh singing is not only much greater than that in Western music but is sometimes greater than the note transition of the phrase.

We used phone segmentation based on forced alignment using hidden Markov model(HMM)-based phone models for speech recognition. Using this method, the onset of a note is the beginning time of its syllable. Moreover, the boundary between the consonant and the vowel in the note can be estimated during the slower start of Noh melody transition, which makes this method suitable. Accurate segmentation requires the preparation of acoustic phone models matched with Noh-Singing signals. However, phonation of Noh singing differs from that of Japanese speech [1], and in preliminary experience, segmentation accuracy is not good when using the acoustic phone model for standard Japanese adult speakers [12].

To prepare a phone model that matches Noh singing, we used the maximum likelihood linear regression (MLLR) speaker adaptation technique [13]. The speaker adaptation technique can fit a general-purpose acoustic model to a specific speaker using only a small amount of data from the target speaker. MLLR speaker adaptation estimates linear transformations for parameters of HMM phone models. In this method, the entire transcription is available from a vocal book. Part of the target data can be used as a supervised adaptation data.

Figure 3 exemplifies the phonetic segmentation result of the pitch contour of the first 4 s of Figure 2. Vertical lines indicate onsets. This figure was plotted using manual labeling for a later explanation.

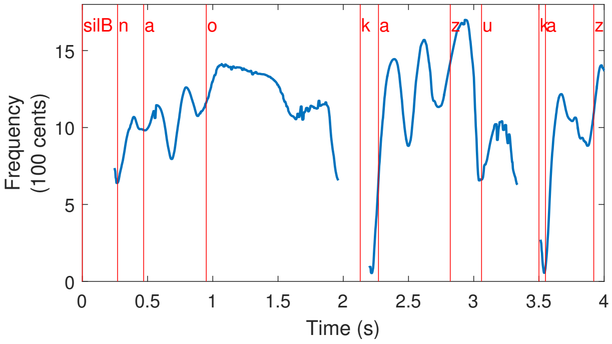


Figure 3. Phone segmentation of pitch contours

E. Melody transcription

Transcribing a melody requires assigning a pitch contour segment to a certain pitch. The relative pitch has to be shifted; however, a shift is not adequate for Noh singing because pitch also changes. A pitch frequency histogram is often used for tonic detection, but, histograms of Noh singing do not have appropriately sharp peaks. Deep vibratos extending wider than the difference between pitches and changeable pitch make the peaks of the histogram broader. Furthermore, unlike Western music, pitch does not change in steps. Therefore, in this study, to transcribe the melody, curves are estimated by polynomial approximation to fit the pitch contour using score information.

This study assumes the following: pitches in a melody are changeable in parallel following the same polynomial coefficients.

1) Initial centered pitch curve: According to the score information and phone segmentation, each phone is assigned one of the three categories: staying, transition, or floating. Using only staying segments, each mean for the pitch in the score (a) is calculated as μ_a. In this step, segments of unvoiced consonants are not used because the pitch estimation may not be accurate.

Then, pitch contour is centered by subtracting μ_a from the pitch contour y. The centered pitch contour y_c is fitted by polynomial regression of degree d. In this study, by preliminary experience, d is determined to be from four to six according to the number of phones Here, let the fitted curve be p. Figure 4 shows the centered contour and the regression curve for the contour of Figure 3. In this phrase, $d = 4$. Figure 4 is plotted in linear scale for frequency.

The pitch for all its segments is not staying, and thus, its mean cannot be calculated. For such pitches, the mean is calculated using the value of the nearest calculated pitch by subtracting the default scale difference value. The scale value is correspondent with 100 cents. For example, in Table I, if μ_{10} was not calculated, μ_{10} is calculated as $\mu_{10} = \mu_6 * 2^{(10-6)/12}$.

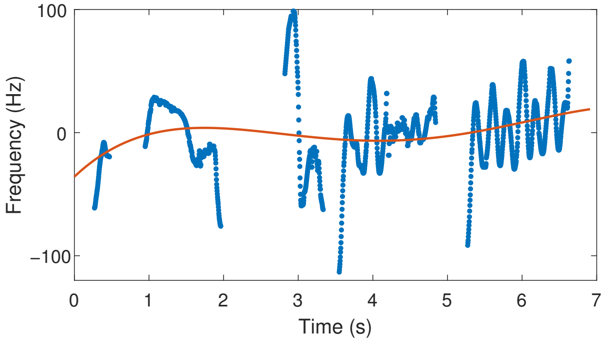


Figure 4. An initial centered pitch contour (dot plot) and an initial centered curve (solid line)

2) Intra-note processing: In Noh singing, note transition time is not strictly determined; hence, if the score directs the pitch to ascend at the i-th note n_i, the pitch ascent delays at the $i + 1$-th note n_{i+1} in the execution [7]. Therefore, in the following process, if necessary, the search range can be extended.

The pitch transition of Noh singing does not start at the beginning of syllables [1]. Adjusting to this knowledge, frames before the beginning of transition are treated the same as staying notes.

a) Floating: Floating is observed as a huge peak in pitch contour. Let the current note be n_i. The initial search range is the vowel region of n_i. For the reason mentioned above, let the search range extend to the next note as $[n_i, n_{i+1}]$. To determine such a peak, first, we subtract the curve corresponding to the pitch of the note in the score a_i from the pitch contour y. Let the result be d. Hence, $d = y - p_i$, $p_i = p + \mu_{a_i}$

Then, let the highest and the nearest of the border between n_i and n_{i+1} peak be a floating transition. Let the intersections between the peak of the pitch contour and p_i be c_1, c_2. The frames before c_1 and after c_2 in the range are treated as staying notes. If there are no intersections for the peak, let the first two nearest local minima of the peak be c_1, c_2. If the pitch contour vanished, let the end point of the contour be c_1, c_2.

b) Transient: Let the current note be n_i. The initial search range is the vowel region of n_i. First, calculate the polynomial curve corresponding to the initial pitch p_{i1} and the final pitch p_{i2}, the same as above p_i. Let the intersections between y and p_{i1} be $s = s_1, \ldots, s_K$, and the intersections between y and p_{i1} be $e = e_1, \ldots, e_L$. If necessary, y is interpolated in search range. Let the first e_m, which is greater than s_k, be the final point. Then, $y(s_k, e_m)$ is fitted by polynomial regression of degree d_t. In this study, d_t is determined as 2. The frames before s_k and after e_m in the range are treated as staying notes.

If there is no s, the search range is extended to the preceding consonant region of n_i. If there is no e, the search range is extended to the next note. Then, s and/or e is calculated again. In addition, if there is also no s, let the first point of the search range be the initial point. Also, if there is no e, let the last point of the search range be the final point.

F. Re-estimation of centered pitch curve and intra note processing

Consequent to intra-note processing, the regions of staying notes are changed. Using this changed data, re-estimate the centered pitch curve. In this step, outliers are ignored using the threshold of three σ because the staying notes' region may include unprescribed ornaments. Then, intra-note processing is executed again.

V. Results and Discussion

Figure 5 shows an example of melody transcription of the pitch contour of Figure 2 using manually labeled phone segmentation. The bullets indicate the onsets of the note. The transcription can be considered as a smooth fit of the graphic notation in Figure 1 to the pitch contour. From this transcription, we can observe the difference between the score and the execution. For example, in the score, the first float is in the first note; however, in the transcription, the float is in the second note. The second and third floats cross the border of the note.

Figure 6 shows the executions of the same phrase by different performers. Both transcriptions share the similar shape but the onsets are different. This sample shows a merit of the proposed method, because such different onsets correspond to different scores in the discrete notation, similar to a score of Western music. In addition, by observing the pitch contours residual to the transcriptions, the vibrato types differ. In the first plot, vibrato is asymmetric to the melodic contour. However, in the second plot, the vibrato is more symmetrical than that in the first. In addition, there is a kind of peak similar to float after the dips at approximately 6 s in both transcriptions. Asymmetric vibrato is one of the most specific characteristics of Noh singing, and is considered as an unprescribed ornament. This suggests that new ornaments will be discovered using this method.

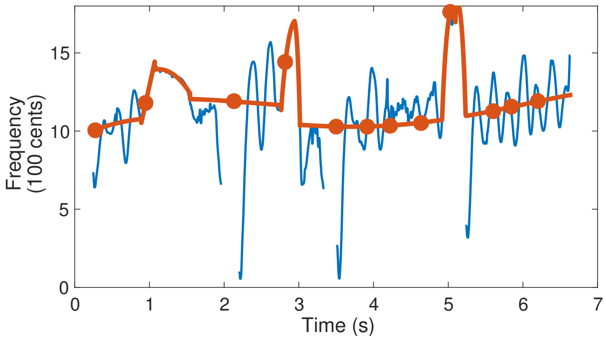


Figure 5. A melody transcription. The blue line is f0 contour. The red line is the melody transcribed with the onset timing using the bullets.

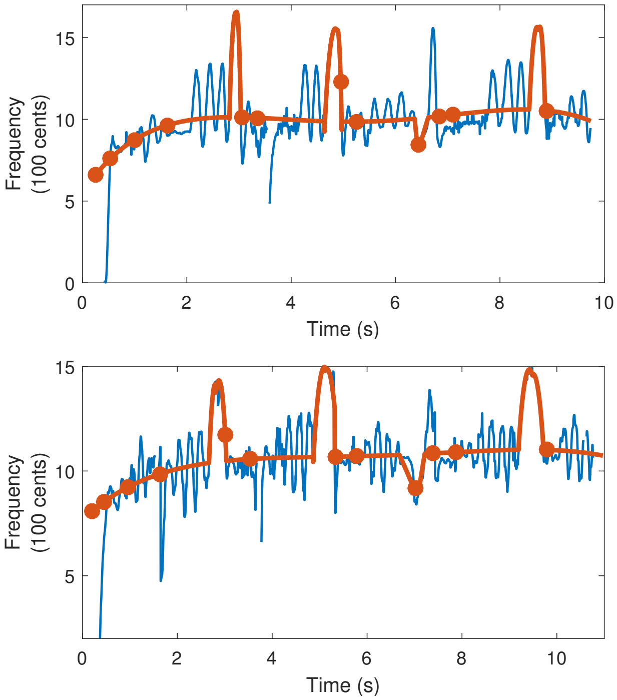


Figure 6. Comparison of the transcriptions of different singers

In Figures 5 and 6, at the beginning of the phrase, pitch contours ascend from much a lower pitch to the target pitch. This ornament is suggested in another commentary [14]. By gaining the degree of the polynomial, transcription better fits such an ornament; however, the gained degree causes an outfit at the other end, there is a lot of literary commentary, which is informal, subjective and ambiguous. Such ornaments are required in quantification, and this transcription will be an aid to that process.

We evaluate the melody transcription accuracy using 17 phrases (from 6 different pieces) in the commentary [7]. In total, we tested 23 audio phrases by four singers. As a general HMM, we used a monophone model trained using read speech of 98 h duration collected from 361 speakers [12]. If a single singer sings an entire target piece, all the data of the target piece can be used as the speaker adaptation data. If multiple singers sing a target piece, two cases will exist. If there is any other pieces that a target singer sings entirely, we can use the adapted model by the data from the piece. If there are no other pieces that a target singer can sing completely, we can use an adapted model of another piece sung by another singer.

Regarding the phone segmentation accuracy, the absolute error compared with the hand-labeled onset was 0.37 s by the general HMM and 0.089 s by the adapted HMMs. Figure 7 shows an example of melody transcription using highly erroneous phone segmentation from the upper pitch contour in Figure 6. The average absolute onset error was 0.23 s. Comparing the data in Figure 6, the first float was missing and the transient just after 6 s was missing. For both cases, successive vowels caused segmentation errors, for example the error of the vowel in the first float was 1.56 s and the error near the transient was 0.39 s, due to the phonation difference in modern Japanese speech. Such vowel differences were reduced by MLLR adaptation.

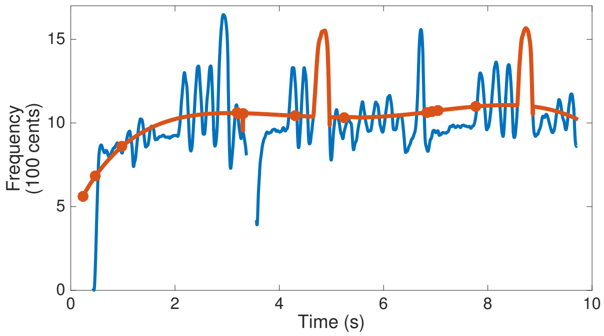


Figure 7. A melody transcription using erroneous segmentation

We evaluated this type of transcription error using the average absolute pitch error and compared it to the transcription estimated using manually-labeled phone segmentation in the cent domain. The average absolute pitch error of the transcription in Figure 7 was 50 cents. The average absolute pitch error was 24 cents by the general HMM segmentation and 13 cents by the adapted HMM segmentation. Figure 8 shows the relationship between the phone segmentation error and the transcription estimation error. The adapted model did not estimate erroneous transcription. For modern or Western music, the onset detection accuracy is evaluated within 50 ms window [10], but for Noh singing more broader window, e.g., 200 ms, might be appropriate for evaluation, because Noh singing is very slow, e.g., the average phone duration was approximately 400 ms in the evaluation data, and the onset is flexible and not as important as it is in Western music.

VI. CONCLUSION

Here, the first melody transcription framework to reflect acoustics for Noh singing is proposed. It was achieved by comparative observations among score information, commentary, and acoustic signals of its multiple executions. The calculation is based on global polynomial regression and modification within the note using score information. According to visual judgment of the plots, the resultant transcription fitted the pitch contour well, and the proposed method's continuous notation was suitable for the flexible nature of Noh singing. Moreover, the potential of discovering new unprescribed ornaments by melody transcription is suggested. For automatic transcription estimation, the speaker adaptation technique using Noh singing audio data as adaptation data was effective.

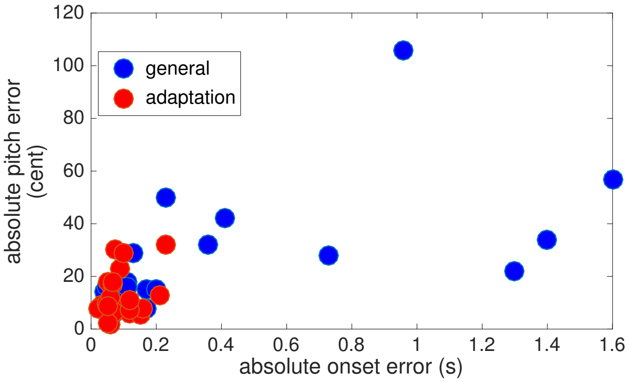


Figure 8. The relationship between the segmentation error and the transcription error

One of the most important remaining issues is evaluation. We are planning to interview professional Noh performers about the concept of melody transcription and the resultant transcriptions.

REFERENCES

[1] I. Nakayama, "Comparison of vocal expressions between Japanese traditional and western classical-style singing, using a common verse," The Journal of the Acoustical Society of Japan, vol. 56, no. 5, 2000, pp.343–348. (in Japanese)

[2] O. Fujimura, et. al., "Noh voice quality," Logopedics Phoniatrics Vocology, vol. 34, 2009, pp. 157–170.

[3] I. Yoshinaga and J. Kong, "Laryngeal Vibratory Behavior in Traditional Noh Singing," Tsinghua Science and Technology, vol. 17, no. 1, 2012, pp. 94–103.

[4] T. Minagawa, "Japanese "Noh" music," Journal of the American Musicological Society, vol. 10, no. 3, 1957, pp. 181–200.

[5] I. Takakuwa, "*Noh/Kyogen utai no hensen*," Hinoki Shoten, Tokyo, Japan, 2015. (in Japanese)

[6] Z. Serper, "*Noh no kotoba no yokuyo*," Engeki Kenkyu, vol. 36, 2013, pp. 51–80. (in Japanese)

[7] K. Miyake, "*Fushi no seikai*(new revised edition)," Hinoki Shoten, Tokyo, Japan, 2012. (in Japanese)

[8] J. Salamon and E. Gómes, "Melody extraction from polyphonic music signals using pitch contour characteristics," IEEE Transactions on Audio, Speech, & Language Processing, vol. 20, no. 6, 2012, pp. 1759–1770.

[9] D. Robinson, "Equal loudness filter," 2015, URL: http://replaygain.hydrogenaud.io/proposal/equal_loudness.html [accessed: 2015-11-25].

[10] J. P. Bello, et. al., "A tutorial on onset detection in music signals," IEEE Transactions on Audio, Speech, & Language Processing, vol. 13, no. 5, 2005, pp. 1035–1047.

[11] N. H. Adams, M. A. Bartsch, J. B. Shifrin, and G. H. Wakefield, "Time series alignment for music information retrieval," Proceedings of ISMIR-04, 2004, pp. 303–310

[12] T. Kawahara, A. Lee, K. Takeda, K. Itou, and K. Shikano, "Recent progress of open-source LVCSR engine Julius and Japanese model repository," Proceedings of ICSLP2004, 2004, pp. 3069–3072

[13] C. L. Leggetter and P. C. Woodland, “Maximum likelihood linear regression for speaker adaptation of continuous density hidden Markov models,” Computer Speech and Language, vol. 9, 1995, pp. 171–185.

[14] M. Yokomichi, “*Nogaku kogi note (utai hen)*,” Hinoki Shoten, Tokyo, Japan, 2013. (in Japanese)

APPENDIX

CD/WMA(WINDOWS MEDIA AUDIO) FILE LIST

- WMF files (All titles are distributed by *Hinoki-Shoten*
  - ○ *Hagoromo*, KANZE Motomasa (solo)
  - ○ *Kiyotsune*, KANZE Motoaki with others
  - ○ *Tamura*, KANZE Motomasa with others
  - ○ *Yashima*, KANZE Motoaki with others
  - ○ *Momijigari*, KANZE Motomasa with others
  - ○ *Hashibenkei*, KANZE Motomasa (solo)
  - ○ *Touboku*, KANZE Motomasa with others
- CDs (*Kanze-ryu Utai Nyumon* CD series (All titles are distributed by kanze.com, sung by KANZE Yoshimasa (solo))
  - ○ *Hagoromo*
  - ○ *Tamura*
  - ○ *Momijigari*
  - ○ *Tsurukame*
  - ○ *Hashibenkei*
  - ○ *Touboku*
- CDs
  - ○ *Hagoromo*, OOE Matasaburo with others, (*Nohgaku Meibankai*)
  - ○ *Kiyotsune*, UMEWAKA Naoyoshi (solo), (*Nohgaku Meibankai*)
  - ○ *Tamura*, KANZE Kiyokazu with others, (*Toei Sound Family*)
  - ○ *Tamura*, FUJINAMI Shigemitsu with others, (Columbia)

VASCO - Mastering the Shoals of Value Stream Mapping

René Berndt, Nelson Silva, Christian Caldera,
Ulrich Krispel, Eva Eggeling
Fraunhofer Austria Research GmbH
Visual Computing
Email: {rene.berndt, nelson.silva, christian.caldera
ulrich.krispel, eva.eggeling}@fraunhofer.at

Alexander Sunk, Thomas Edtmayr, Wilfried Sihn
Fraunhofer Austria Research GmbH
Production and Logistics Management
Email: {alexander.sunk, thomas.edtmayr, wilfried.sihn}
@fraunhofer.at

Dieter W. Fellner
Institute of ComputerGraphics and KnowledgeVisualization (CGV), TU Graz, Austria
GRIS, TU Darmstadt & Fraunhofer IGD, Darmstadt, Germany
Email: d.fellner@igd.fraunhofer.de

***Abstract*—*Value stream mapping* is a lean management method for analyzing and optimizing a series of events for production or services. Even today the first step in value stream analysis - the acquisition of the current state - is still created using pen & paper by physically visiting the production place. We capture a digital representation of how manufacturing processes look like in reality. The manufacturing processes can be represented and efficiently analyzed for future production planning by using a meta description together with a dependency graph. With our Value Stream Creator and explOrer (VASCO) we present a tool, which contributes to all parts of value stream analysis - from data acquisition, over planning, comparison with previous realities, up to simulation of future possible states.**

Keywords*–*Value stream mapping; lean management; content authoring.

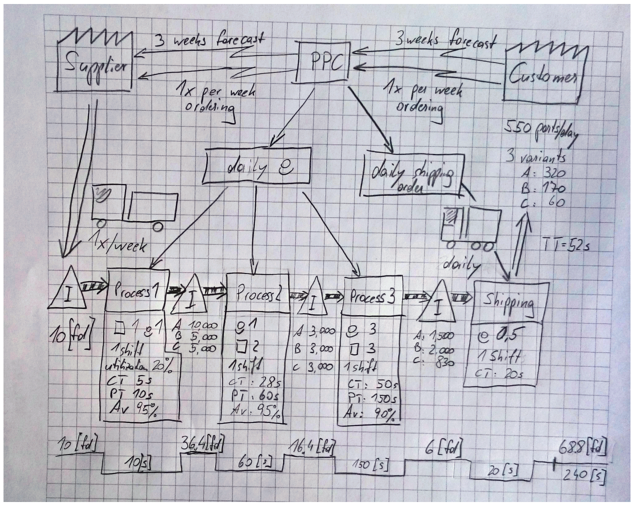


Figure 1. Typical hand-drawn VSM diagram created during a shop floor acquisition in a production facility.

I. Introduction

Value Stream Mapping (VSM) is an abstract lean manufacturing technique for optimizing the material and information flows from production up to the delivery of products to the customers. Usually, this is done by drawing current and future state maps by hand, allowing the optimization of production by identifying bottlenecks and wastes. Figure 1 shows a typical hand-drawn board template for data acquisition at the "shop-floor". The concepts of VSM are usually represented by a set of standard symbols, which got various properties attached. Typical properties, e.g., for a VSM process (which represents a production step like welding or assembly) include information about the process time, scrap rate, workers involved in the production, but can also contain data about published enhancements of traditional VSM, e.g., space useage for production and logistics, transport distance and transport time [1].

The history of designing process maps and flowcharts to represent the flows of materials and information in a factory can be traced at least back to 1915, where in a book by Charles E. Knoeppel entitled "Installing Efficiency Methods" we can find interesting graphical representations about the processes and routings in a manufacturing plant [2].

Nowadays, value stream mapping with traditional pen & paper method faces new challenges in practical utilization [3]. To prevent incorrect application, it is necessary to have company wide standards for drawing, data collection and analyzing current state maps. Therefore, VASCO was established to close this gap for enabling the planning of successful future state value stream maps in a digital manner.

The next section will give an overview of the related work and programs which inspired the creation of VASCO. Section 3 shows the main functionalities of VASCO, how VSM diagrams are modelled within the system and and how the automatic calculations are handled. The last section concludes our work and will give an outlook of further features.

II. Related Work

VSM was originally developed as a method within the Toyota Production System [4][5] and introduced as a distinct methodology by Rother & Shook [6]. VSM is a simple, yet very effective, method to gain a holistic overview of the conditions of the value streams within a production environment. Based on the analysis of the current state maps, flow-oriented future state maps are planned and implemented [6][7][8].

A value stream includes all activities, i.e. value adding, non-value adding and supporting activities that are necessary

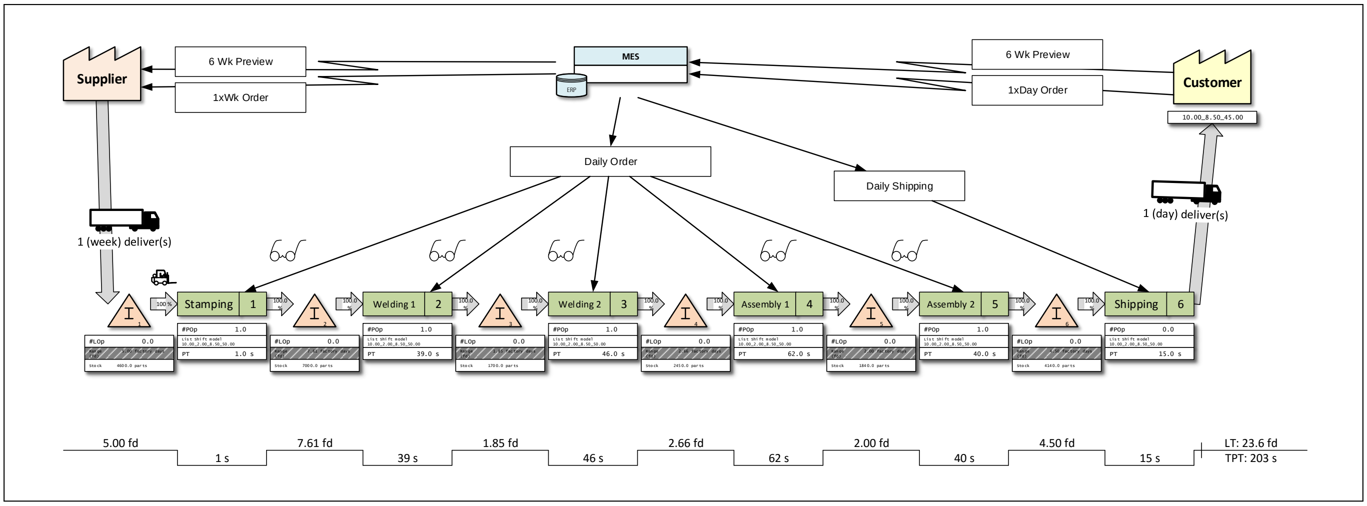


Figure 2. A typical VSM diagram, describing the value chain in production from the supplier to the customer.

to create a product (or to render a service) and to make it available to the customer. This includes the operational processes, the flow of material between the processes, all control and steering activities and also the flow of information [3]. In order to assess possible improvement potential, VSM considers, in particular, the entire process time (sum of time of all production steps) compared with the overall lead time (time from the customer ordering to the moment of delivery). The greater the distinction between operating time and lead time the higher the improvement potential [7].

Value stream simulation can be used in lean manufacturing to support the optimization of production. It allows an early stage insight into productivity, effectiveness and service level without the need of creating very detailed and time consuming simulation models. This means that a simulation in lean management workshops can now be done by lean experts instead of relying in simulation experts. Traditionally VSM is a pen & paper tool that captures the state of the system at the state it was drawn. Component based modeling divides the simulation into a set of simulation blocks [9]. These blocks can be used to create value stream maps that are generic and reusable. In our application VASCO, we also support several reusable blocks that allow the user to easily create value stream diagrams that are reusable in a standardized way. By utilizing standard simulation building blocks one can easily know the state of the system under different circumstances allowing for better decision making. Another important aspect of using value stream maps and specially in a digital form, is that the production and delivery processes are optimized from the customers' point of view [6].

VSM is also seen as a tool to show the outcomes in a shorter period of time at minimal costs. The lean consultants can now represent and capture the current state of the process at a certain state and time and start projecting the future proposed state of the value stream. Based on lean concepts the two states can be simulated and key measurements are assessed. These simulation results can easily demonstrate the improvements [10]. In manufacturing, there are three types of operations that are undertaken to represent a type of waste that might occur: non-value adding, necessary but non-value adding and value-adding operations [11]. The first type is pure waste with unnecessary actions that should be completely eliminated. The second type involves actions that are necessary but might be wasteful. The third type are value-adding operations representing processes that convert raw materials into finished products.

The capture of information to a digital form is often not sufficient. From the point of view of using a digital tool to capture the state of a process, there are several applications that can be used and are available. However, in their paper, Shararah et al. [9] introduce the Value Stream Map Simulator using ExtendSim (VSMSx) as a powerful tool designed to facilitate the implementation of lean manufacturing by simulating the value stream map through standardized simulation building blocks. The company Siemens created as part of their Product Lifecycle Management - PLM product line, an optional extension library called Plant Simulation Value Stream Mapping Library [12]. The company immediately reported productivity increases by as much as 20 percent and improvements of 60 percent related with the reduction of inventories and cycle time. Other benefits such as investment risk reduction (through early feasibility analysis capabilities), better line planning and allocation and significant increases in the resource utilization were also highlighted. The capability of being able to define what-if scenarios without disturbing existing production systems during the planning process is pointed as one of the most important features of any VSM planning tool. Plant simulation is also referred as an important feature of such systems, because it facilitates the comprehension of complex production systems and processes. Resource utilization, material flows and supply chains maybe therefore optimized. The question of ”Why perform value stream mapping in Plant Simulation?” is also debated in this technical report. Factors such as the reduction of cost for data collection by reducing the number of objects describing the processes (by utilizing pre-defined logic blocks) or the reduction in analysis effort through automated analysis modules have an important role in deciding to use VSM. In order highlight the dynamic effects (which remain hidden in the static paper based mapping of the value chain), a digital representation (through computer simulation) of the value stream is required.

According to Nash & Poling [13], the value stream mapping has some disadvantages associated with it. It points to the fact that originally, VSM did not include any significant monetary measure for value. It is the stakeholders responsibility to determine determine which activity can be marked as value as well as which activity can be marked as waste. The task of decision-finding may take a lot of valuable time.

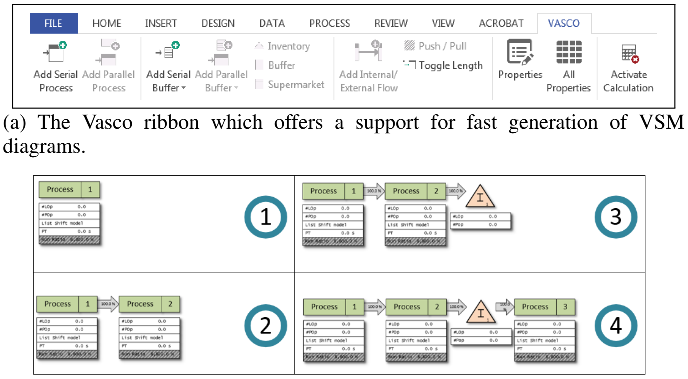
Another important challenge arises from the fact that there is the need to not only capture data and information about the processes and the information flows involved, but also it is beneficial to have a digital representation of how these processes look like in reality [14], in fact ultimately we would like to achieve what is sometimes called ”The Digital Twin Concept Model” [15]. Similarly, in our approach we are taking the steps necessary to provide this type of vision. When we analyze the current arrangement of an assembly line and we capture this information on a VSM diagram (current state). At a later stage we do not want to come back to the production area to visual re-check the arrangement of machines, workers, to discover how are the parts actually delivered and stored or to know what are the space constrains to be able to describe and demonstrate how the actual work of the existent implemented processes is being performed. To have a better view of what should be improved when preparing the future state VSM, it is desirable that the new digital tool for the creation of VSMs can allow the users to capture and then to find annotations in the form of pictures, videos or 3D representations of the past, current and future reality of the production sites. Therefore, every time a user is handling a VSM diagram, he will be able at any step of the process to access these digital catalog of the different processes, that are now linked to the VSM digital representation.

A field research on available standard software tools showed a lack in possibility of detailed analysis. While some tools just provide basic drawing aids for creating value stream maps (e.g. Microsoft Visio [16]), other tools like iGrafx [17], Plant Simulation [18] or SmartDraw [19] also support lead time calculations and basic simulation. None of them considers the availability of data in production lines, which is a big deal nowadays in order to cope with all the complexity and achieve transparency. Nevertheless, detailed analysis and transparency of value streams are needed to reveal improvement potentials.

To address the challenges in mastering the increasing complexity in the VSM data models, we are developing a highly customizable tool for authoring and managing value streams. The next section gives an insight on the key features of VASCO.

III. VASCO MAIN FUNCTIONALITES

VASCO is implemented as a Microsoft Visio Plugin. This allows to reuse the drawing and connecting shapes functionality already provided by Visio. A VASCO value stream diagram can be combined with other shapes and features included by Visio or other 3rd-party AddIns. One important aspect in the design of the VASCO focuses on the user experience. Figure 3(a) shows the ribbon toolbar for VASCO. All available control elements are optimized for the fast creation of VSM diagrams, especially adding and positioning process or buffer symbol. Typical repeating tasks are automated like the adding of serial process/buffer symbols, where VASCO already connects the two symbols using an internal flow connector. The inserted process/buffer stays selected, so that the user can immediately use the commands ”Add serial process”/”Add serial buffer” multiple times. Figure 3(b) shows how to create a VSM diagram. From (1) to (4) using only mouse clicks - or if you are on a touch device, then only 4 touch events are needed, which is much faster then a hand-made drawing. Therefore, the usual manual steps of transforming the hand-made drawings into digital documents is now completely obsolete when using VASCO.



(a) The Vasco ribbon which offers a support for fast generation of VSM diagrams.

(b) Drawing a simple VSM.

Figure 3. With minimum user interaction the drawing process in Figure 3(b) can be achieved with our Vasco toolbar shown in Figure 3(a)

A. Definition of VASCO symbols

As seen in figure 2 a value stream consists of a variety of standardized shapes and information. VASCO adds properties and the calculation logic to the VSM shapes to the main VSM symbols:

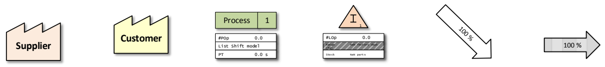
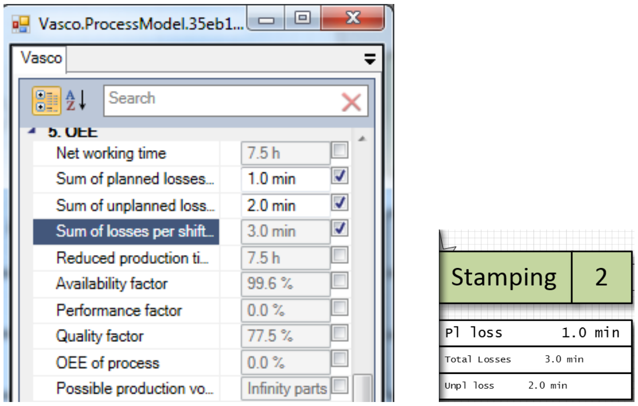


Figure 4. The standard VSM symbols (from left to right): Supplier, Customer, Process, Buffer, External flow, Internal flow.

- The **supplier** is the manufacturer which ships the goods into the factory.
- The **customer** is a company, merchant or another entity who orders goods and requires them to be shipped regularly. The customer determines the demand and the resulting takt time, which is a key value driving almost all calculations within a value stream.
- The **process** is a step, which adds value to goods by altering or modifying it.
- The **buffer** is an intermediate step where the factory goods are stored. This storage might be an input for the next process, a general depot for delivering goods to the customer or from the supplier.
- The **external flow** connects a supplier or a customer with a buffer or a process.
- The **internal flow** connects buffers and processes with each other. The main difference between an external



Vasco.ProcessModel.35eb1...

Vasco

Search

5. OEE

Net working time	7.5 h	□
Sum of planned losses...	1.0 min	☑
Sum of unplanned loss...	2.0 min	☑
Sum of losses per shift...	3.0 min	☑
Reduced production ti...	7.5 h	□
Availability factor	99.6 %	□
Performance factor	0.0 %	□
Quality factor	77.5 %	□
OEE of process	0.0 %	□
Possible production vo...	Infinity parts	□

Stamping	2

Pl loss	1.0 min
Total Losses	3.0 min
Unpl loss	2.0 min

(a) The complete property model of the stamping process.

(b) Selected attributes of the property model displayed below the process symbol. 5(a).

Figure 5. The properties of the stamping process and how the process is displayed on the VASCO sheet.

and an internal flow is that a internal flow might connect a process with multiple buffers or processes. This means that the internal flow is augmented with a property called material spreading which defines how many percent of material spreads from one process or buffer to their successors.

A valid *value stream graph* consists of one supplier at the start, one or more customers at the end, one or multiple processes, and zero or multiple buffers. Each of the processes and buffers are connected with internal flows and two external flows connecting the supplier and customer with the network.

B. Intelligent Symbols

One of the significant features of VASCO is that all symbols defined in Section III-A, are fully customizable by a configuration file. This configuration file defines which properties are added to the symbol. These properties can be classified into two major categories: manual input values or calculated values. The manual values are entered by the user, whereas the calculated values depend on a formula consisting of manually entered or other calculated values. The formula definition is also part of the configuration file and can be modified even at run-time.

For example, let's have a process which drills a hole into a plate. This process has two manual input values. The value adding time which is the time period when the drill actually drills the hole and the setup time. The setup time is the time period needed for positioning the drill and the time the assembly line needs to bring the plate into position. With these two values in the process it is possible to add a third value, the process time which automatically calculates the sum of the setup time and the value adding time. This automatic calculated value can be displayed on a databox in the process shape (see Figure 5). If a value of these properties changes, all dependent values are updated immediately (see Figure 5(a)).

One special case of calculated values is when the values not only depend on the local process, like in the example above, but also in other places from other symbols, like the following processes. These special calculations are called graph calculations. The graph calculations are also defined in the configuration file, but require a complete value stream graph in order to perform their calculations. For this, VASCO has two different modes. The first mode, is the design mode. In this mode, the user can add processes, buffers and connect them with each other. The calculations which are only local are calculated in this mode. The second mode is the calculation mode, where all graph calculations calculate their value. In this mode it is not possible to add, remove or connect symbols with each other.

To get a better picture about the calculation mode consider a customer who requires 100 items. We have 3 processes which are connected in series. Each process has a scrap rate of 10%. Now each of the processes has to accommodate the scrap rate of the following processes and produce more goods. That in the end the customer gets his 100 items. Therefore, in our example the first process requires 139 items. This example can become arbitrarily complex with parallel processes and the material spreading in internal flows. In the calculation mode all values are live updated and displayed. So if the customer requests that the factory delivers more items, it is then immediately visible how many more raw material the first process requires. This is also the reason why it is not allowed to edit the path, remove or add further processes during the calculation mode, as all values would be invalid with a unfinished value stream graph.

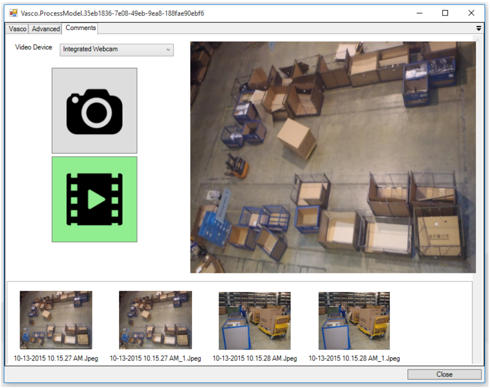
When a user adds a new intelligent symbol to a diagram, e.g., a buffer, this symbol becomes now automatically the current selected symbol. This allows the automation of the possible next choices for symbols that can be added to the diagram (connected to the current symbol). In this way, when the user looks to the application main toolbar, only symbols that are possible to be connected to the previous symbol, are available for a next drop in the diagram. When the user intent is to connect two symbols, e.g., the user wants to connect a buffer with a process, the user pre-selects these two symbols. After this step, the application automatically highlights the possible connections that can be added between the selected symbols. The users reported that these methods significantly improve the productivity and the usability of our application interface. These and other improvements will be the target of future studies, where we will access the overall usability of the tool and compare it with other existent VSM applications.

C. Key Performance Indicators and Data Lines

As referred in the related work section, an important aspect in the analysis of VSM diagrams is the extraction and automatic calculation of *Key Performance Indicators* (KPI).

Key performance indicators can be calculated local, e.g., for a single process (e.g. OEE rate) or buffer (e.g. local lead time) but also for the whole graph /value stream (e.g. total lead time). These values are calculated automatically and are visualized in several data lines below the drawn value stream. As an example for the several supported data lines in VASCO, the time line consisting of total process time and lead time is shown in Figure 2.

When discussing with the main key holders (manufacturing and production consultants or VSM and processes simulation owners) involved in the event of capturing processes and information flows (as well as many other related information

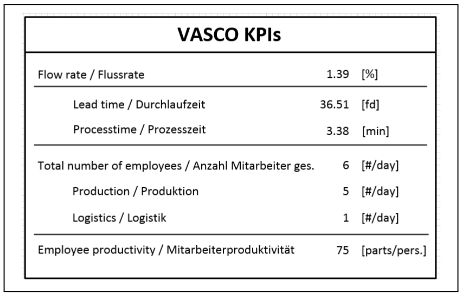


Vasco.ProcessModel.35eb1836-7e08-49eb-9ea8-188fae90ebf6

Video Device: Integrated Webcam

10-13-2015 10.15.27 AM.Jpeg
10-13-2015 10.15.27 AM_1.Jpeg
10-13-2015 10.15.28 AM.Jpeg
10-13-2015 10.15.28 AM_1.Jpeg

Figure 6. The Comments plugin allows to annotate VSM symbol with photos and videos taken from the camera of a tablet.



VASCO KPIs

Flow rate / Flussrate	1.39	[%]
Lead time / Durchlaufzeit	36.51	[fd]
Processtime / Prozesszeit	3.38	[min]
Total number of employees / Anzahl Mitarbeiter ges.	6	[#/day]
Production / Produktion	5	[#/day]
Logistics / Logistik	1	[#/day]
Employee productivity / Mitarbeiterproduktivität	75	[parts/pers.]

Figure 7. The Key Performance Indicator giving an overview of the operating numbers in the factory.

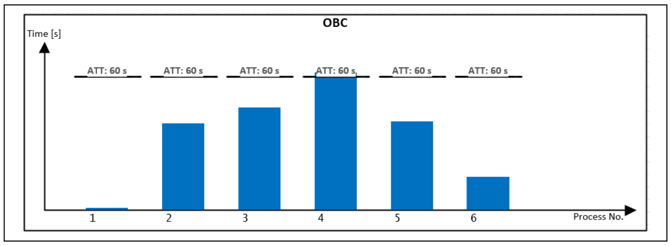


Figure 8. OBC showing the process time in relation to the takt time.

captured now in a digital form), one of the most desired features is the capability of calculating improved business metrics. These metrics allow the evaluation of factors that are critical to the success of an organization. In our tool, we calculate and present metrics that are related with human resources, costs, performance and workload balancing management. These are essential to the reduction of costs and to the improvement of performance of processes and persons.

We present results to the users in a concise way through resume maps in each step of the calculations procedures. This is a realistic and simple way to digitally represent the past, current and future reality inside all manufacturing sites.

D. Extensibility

One key-feature of VASCO is extensibility. While self being an Microsoft Visio addin, it can be customized by plugins itself. The basic version of VASCO is already shipped with three plugins, extending the basic functionality of the tool:

- **Comment-Plugin** VASCO was designed to make the acquisition and calculation of a new value stream easier and to replace the pen & paper acquisition. With the pen & paper method it is always possible to add different comments to the different symbols. In order to give the VASCO user a similar feature during the acquisition a comment plugin was created. This comment plugin enhances every symbols on a VASCO page with a comment tab (see Figure 6). When we observed during the data acquisition process that users sometimes only copied key figures from a machine into this comment tab, we further enhanced the comment-plugin with a snapshot ability. With this snapshot ability the user doesn't need to copy the values himself. The user only has to take a snapshot with the tablet. It is also possible to record a video with the comment plugin. This can be done to record different views of the machine or to record the voice of the person who does the acquisition so that there is not even the need to write textual facts in the comments box.
- **KPI-Plugin** The KPI-Plugin adds an additional visual features (see Figure 7) to the Visio page. This shape displays the key performance indicators of the factory in a clear fashion. Once a VASCO graph is complete and VASCO itself is in calculation mode, the values are calculated and automatically updated when a value in the graph changes.
- **OBC-Plugin** The operator balance chart (OBC) visualizes the total amount of work of each process compared to the takt time. An OBC supports the critical task of redistributing work elements among operators. This is essential for minimizing the number of operators needed by making the amount of work for each operator very nearly equal to, but slightly less than, takt time [20]. Figure 8 shows the OBC chart of the given example.

E. Data Model and Calculations

This section describes the data model used in VASCO, and discusses some implementation aspects of the evaluation.

1) Graph Structure: Naturally, the elements of a value stream map can be represented using a graph structure. A graph $G = (N, E)$ consists of a set of nodes N together with a binary relation E on the set. Each concept of a VSM (e.g. a process or a flow) is represented by a node $n \in N$ in the graph, connections between VSM concepts are represented by a directed edge $e \in E$ between the corresponding nodes.

Each node, or concept, contains a set of named properties. These properties can either be set to a constant value, or can be calculated from other properties.

2) Evaluation of Calculated Values: Our key observation was that the dependencies of calculated values need to be represented in the data model of the system for an efficient evaluation. Therefore, VASCO contains a second graph structure that represents the dependencies of calculated values. This allows an efficient re-evaluation if the user changes a property value, as only dependent values will be recalculated.

A calculated value may depend on values from other nodes in the graph, in this case the concrete dependency

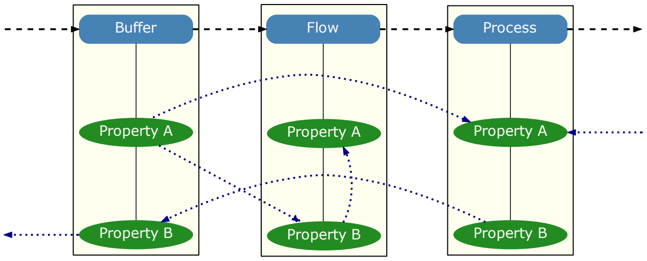


Figure 9. The data model of VASCO captures the structure of the VSM graph.

graph depends of the VSM neighborhood of the concept node containing the calculated value, therefore changes in the VSM graph structure also entail changes in the dependency graph.

An example of this graph structure can be seen in Figure 9. In this example the connection between Buffer, Flow and Process are shown with black dashed lines. Furthermore, the dependencies of calculated Properties are represented by a second graph (blue dotted lines), e.g. property A of Buffer depends on property B of Flow and property A of Process. VASCO contains a meta-representation of VSM objects and the dependencies of their calculated properties such that the dependency graph will be automatically created from the VSM graph.

IV. Future Work & Conclusion

With VASCO it is possible to use only one tool throughout the complete work-flow of value stream analysis. It dramatically simplifies the data acquisition at the shop-floor and offers a large tool set for analyzing and improving the production/logistic value chain.

Also the extraction of production metrics can be done at any stage of the work-flow creation, allowing the users to immediately have calculations feedback about the impact of their changes when designing future state maps. Lastly, by using our tool the users can capture information along the entire value stream analysis process, starting with visits to the production sites, where the users can capture images and videos of the working processes as side annotations, up to the creation of a new diagram based on previous processes workflow states with comparisons between multiple state realities of the existent manufacturing processes, where the users can still access all the annotated information about old and current manufacturing processes.

Future work will concentrate on combining and integrating VASCO with other professional simulation tools. This allows to simulate various combinations and new arrangements for the future state of the VSM. Another important aspect is the integration of sustainability criteria within a VSM, that will significantly help manage and reduce the amount of waste, resulting from the manufacturing processes. This will also provide new metrics and KPI's that help to capture each company production reality in a digital way.

User experience is always a primal focus in all industrial applications. We are planning experiments where the users will perform fundamental tasks with our tool. With the help of an eye tracker equipment, we will record data about the way users perform their tasks and about their individual preferences.

It is our intention to assess how our tool is used in reality by the final users and to study its usability. We expect to be able to use this data to improve the overall experience of the users and as a way to boost the productivity of the users when working with our tool.

References

[1] P. Kuhlang, T. Edtmayr, and W. Sihn, "Methodical approach to increase productivity and reduce lead time in assembly and production-logistic processes," CIRP Journal of Manufacturing Science and Technology, vol. 4, no. 1, 2011, pp. 24–32.

[2] C. E. Knoeppel, Installing efficiency methods. New York, The Engineering Magazine, 1915.

[3] A. J. D. Forno, F. A. Pereira, F. A. Forcellini, and L. M. Kipper, "Value Stream Mapping: a study about the problems and challenges found in the literature from the past 15 years about application of Lean tools," The International Journal of Advanced Manufacturing Technology, vol. 72, no. 5-8, Feb. 2014, pp. 779–790.

[4] T. Ohno, Toyota Production System: Beyond Large-Scale Production. Productivity Press, Mar. 1988.

[5] J. K. Liker, The Toyota Way: 14 Management Principles from the World's Greatest Manufacturer. Munich: McGraw-Hill, 2004.

[6] M. Rother and J. Shook, Learning to see: value stream mapping to add value and eliminate muda. Cambridge: Lean Enterprise Institute, 2003.

[7] K. Erlach, Wertstromdesign. Berlin, Heidelberg: Springer Berlin Heidelberg, 2010.

[8] T. Klevers, Wertstrom-Mapping und Wertstrom-Design. Verschwendung vermeiden - Wertschöpfung steigern, 1st ed. Landsberg am Lech: mi-Fachverlag, Jan. 2007.

[9] M. Shararah, K. El-Kilany, and A. El-Sayed, "Component based modeling and simulation of value stream mapping for lean production systems," Proc. of FAIM Conference, 2010, pp. 881–888.

[10] M. Ravesh, "Confirming lean transformation outcomes by using simulation," Proceedings of the Asia Pacific Industrial Engineering & Management Systems Conference, 2012, pp. 1732–1736.

[11] Y. Monden, Toyota Production System: An Integrated Approach to Just-In-Time, 4th Edition. CRC Press, 2012.

[12] SIEMENS, "Tecnomatix plant simulation: Value stream mapping," SIEMENS, Tech. Rep., 2012.

[13] M. A. Nash and S. R. Poling, Mapping the Total Value Stream: A Comprehensive Guide for Production and Transactional Processes, 1st Edition. CRC Press, 2008.

[14] T. Rohrlack. A bentley solution paper for automobile manufacturers: The digital factory from concept to reality. [Online]. Available: http://ftp2.bentley.com/dist/collateral/whitepaper/DFWhitepaper.pdf [retrieved: January, 2016]

[15] M. Grieves. Digital twin: Manufacturing excellence through virtual factory replication. LLC. (2014)

[16] Microsoft Corporation, "Professional Flow Chart and Diagram Software — Microsoft Visio," www.visiotoolbox.com, 2016, [Online; retrieved: January, 2016].

[17] iGrafx, LLC, "iGrafx Flowcharter," http://www.igrafx.com/, 2016, [Online; retrieved: January, 2016].

[18] Cards PLM Solutions B.V., "Tecnomatix Plant Simulation," http://www.cardsplmsolutions.nl/en/plm-software/tecnomatix/plant-simulation-6, 2016, [Online; retrieved: January, 2016].

[19] SmartDraw, LLC, "SmartDraw Value Stream Mapping," http://www.smartdraw.com/value-stream-map/, 2016, [Online; retrieved: January, 2016].

[20] Lean Lexicon 5th Edition. Berlin, Heidelberg: Lean Enterprise Institute, Inc., 2010.