Optimized Positioning of Autonomous Surgical Lamps

Jörn Teuber[a], Rene Weller[a], Ron Kikinis[a], Karl-Jürgen Oldhafer[b], Michael J. Lipp[b], and Gabriel Zachmann[a]

[a]University of Bremen, Bremen, Germany
[b]Asklepios Klinik Barmbek, Hamburg, Germany

ABSTRACT

We consider the problem of finding automatically optimal positions of surgical lamps throughout the whole surgical procedure, where we assume that future lamps could be robotized. We propose a two-tiered optimization technique for the real-time autonomous positioning of those robotized surgical lamps. Typically, finding optimal positions for surgical lamps is a multi-dimensional problem with several, in part conflicting, objectives, such as optimal lighting conditions at every point in time while minimizing the movement of the lamps in order to avoid distractions of the surgeon. Consequently, we use multi-objective optimization (MOO) to find optimal positions in real-time during the entire surgery. Due to the conflicting objectives, there is usually not a *single* optimal solution for such kinds of problems, but a set of solutions that realizes a Pareto-front. When our algorithm selects a solution from this set it additionally has to consider the individual preferences of the surgeon. This is a highly non-trivial task because the relationship between the solution and the parameters is not obvious. We have developed a novel meta-optimization that considers exactly this challenge. It delivers an easy to understand set of presets for the parameters and allows a balance between the lamp movement and lamp obstruction. This meta-optimization can be pre-computed for different kinds of operations and it then used by our online optimization for the selection of the appropriate Pareto solution. Both optimization approaches use data obtained by a depth camera that captures the surgical site but also the environment around the operating table. We have evaluated our algorithms with data recorded during a real open abdominal surgery. It is available for use for scientific purposes. The results show that our meta-optimization produces viable parameter sets for different parts of an intervention even when trained on a small portion of it.

Keywords: depth camera, open surgery, operating site, optimal lighting, operating site lighting

1. INTRODUCTION

Perfect lighting conditions are essential in surgery. According to basic guidelines for surgical luminaire systems,[1] lamps must be able to provide illumination of at least 40,000 lux at the surgical site. Moreover, all lamps together should not exceed an illuminance of 160,000 lux. While the pure illuminance intensity can be easily achieved by modern surgical lamps, the positioning of the lamps to guarantee optimal lighting conditions throughout the whole operation is still a challenge, especially in highly dynamic interventions such as open abdominal surgery. Usually, a set of surgical lamps is mounted above the operating table and the surgeon or the surgical staff have to adjust them manually. Even in highly advanced and experimental operation rooms, the surgical lamps are still moved this way. This re-positioning interrupts the workflow[2] and as a result, that was reported by surgeons and scrub nurses, the surgeons tend to carry on working under sub-optimal lighting because they don't want to interrupt their current task. Moreover, the necessity to manually adjust the lamps facilitates the need for low-hanging lamps with sterile handles, which have to be replaced before every surgery to maintain sterility.

In our previous work,[3] we introduced a generalized pipeline for autonomous surgical lamps. The pipeline generates several *workspace maps* — 2-dimensional representations of all the lamps' working area — which are sufficient to evaluate the current conditions in the operating room and find positions for a set of lamps based on current and past occlusion. So effectively, our pipeline reduces the problem of lamp positioning to two dimensions.

Further author information: (Send correspondence to Jörn Teuber)
Jörn Teuber: E-mail: jteuber@cs.uni-bremen.de, Telephone: +49 421 218 63998
Gabriel Zachmann: E-mail: zach@cs.uni-bremen.de, Telephone: +49 421 218 63991

In this submission, we present an improvement and an extension to the part of the pipeline that is responsible for optimizing the positions of the lamps. This includes the decision to keep the lamps at their current place if suitable. This optimization is one of the most critical parts of the system, as it determines the success or failure of the autonomous lamps to provide optimal lighting.

The optimization has to be able to provide a solution with several different and conflicting goals in mind. First and foremost, the lamps must never collide with each other and the equipment in the operating room. Moreover, they should avoid occlusions between themselves and the surgical site. Finally, and conflicting with the previous goals, the system should minimize the lamps' movement to reduce possible distractions for the surgeons. This constraint favors an optimization that remains reasonably occlusion-free over longer stretches of time.

1.1 Contribution

In detail, our contributions in this paper are:

- A two-tiered optimization approach for completely autonomous surgical lamps consisting of:
  - A runtime optimization using the scalarization approach with a weighted sum, running in real time on the incoming data and
  - A meta-optimization using the pareto approach of multi-objective optimization to identify the best weights for the first optimization. The results of this optimization are sets of weights paired with easy to understand metrics representing the performance of the autonomous lamps on the used recording of a surgery.
- An evaluation of the optimization done on a recording of a real-life surgery. We made this data available to the scientific community for other scientific purposes.

2. METHODS

We start with a short recap of our generalized pipeline for autonomous lamps before we explain our novel optimization methods 2.2.

2.1 Our Optimization Pipeline

The basic idea for our autonomous surgical lamps is that they are freely movable on at least a 2-dimensional area above the operating table. In practice, this can be realized by replacing the typical hinged brackets of the surgical lamps with motorized units or using complete robotic arms. To find optimal positions for the surgical lamps, we need to capture the current spatial distribution of possible occluders, i.e. we need a 3-dimensional representation or model of the area around the operating table. We obtain this model by using a depth camera that is mounted 1.5-2 meters directly above the operating table, looking down. This ensures that we capture enough information about the occluders as well as get a decent look onto the surgical site for use in future projects, like hand- and site-tracking.

The image stream of the depth camera is the starting point of our pipeline (see Figure 1). The pipeline's stages can be roughly subdivided into four general stages. The first stage is gathering information about the environment around the operating table to create the aforementioned model. In addition to the depth image stream from the depth camera, this includes a simple temporal filter on the depth data to reduce the noise, and the transformation of the depth images to 3-dimensional point clouds.

In the second stage, we use perspective projection to project the previously obtained model, i.e. the point cloud, with standard computer graphics techniques from the point of view of the current operating site, looking up, onto the working space of the lamps (see Figure 2). This yields an occlusion map of the lamps workspace, which stores for each pixel whether an occlusion between the surgical site and the corresponding spot on the lamps workspace on the ceiling occurred or not. This reduces the problem of finding an optimal position in a 3D space to only two dimensions. Basically, the occlusion map already enables us to search for currently occlusion-free positions for the lamps.

In the third stage, this occlusion map is further processed into four different maps (see Figure 3). The first map stores the discrete distances to the nearest definite occlusion. This map is useful to quickly determine the

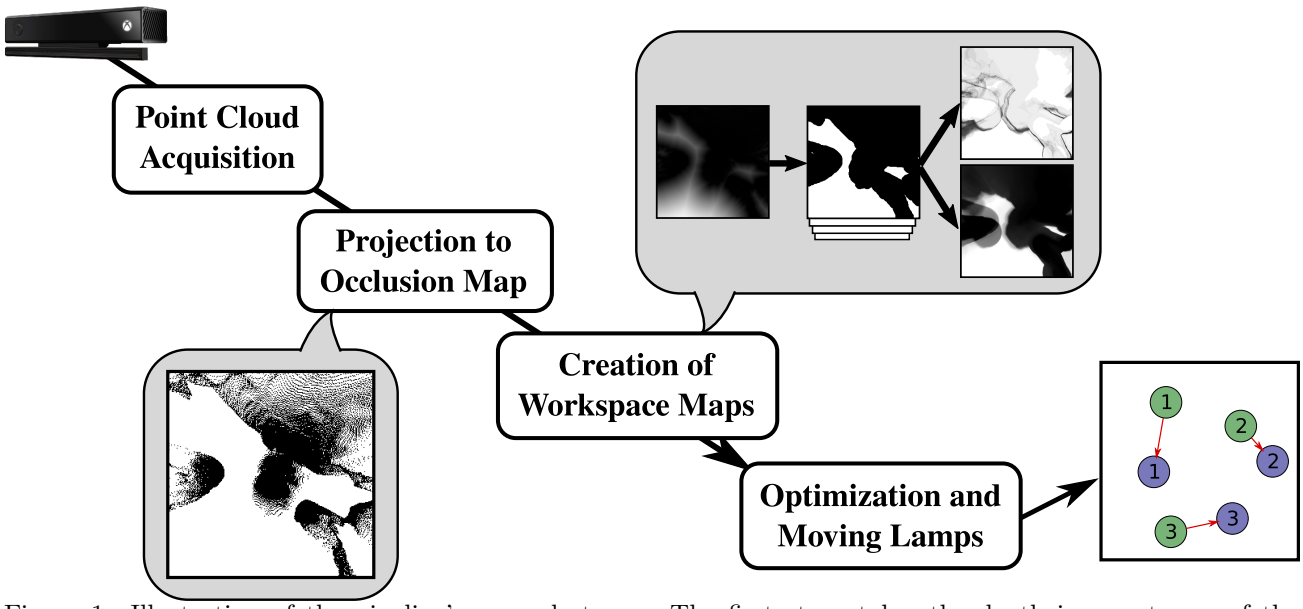

Figure 1: Illustration of the pipeline's general stages. The first stage takes the depth image stream of the connected depth camera (in this case a Kinect v2) and transforms it into a point cloud. The second stage projects the point cloud onto the workspace, yielding the occlusion map which is processed in the third stage into the four workspace maps. The forth and last stage optimizes based on these maps and moves the lamps towards the new optimal positions.

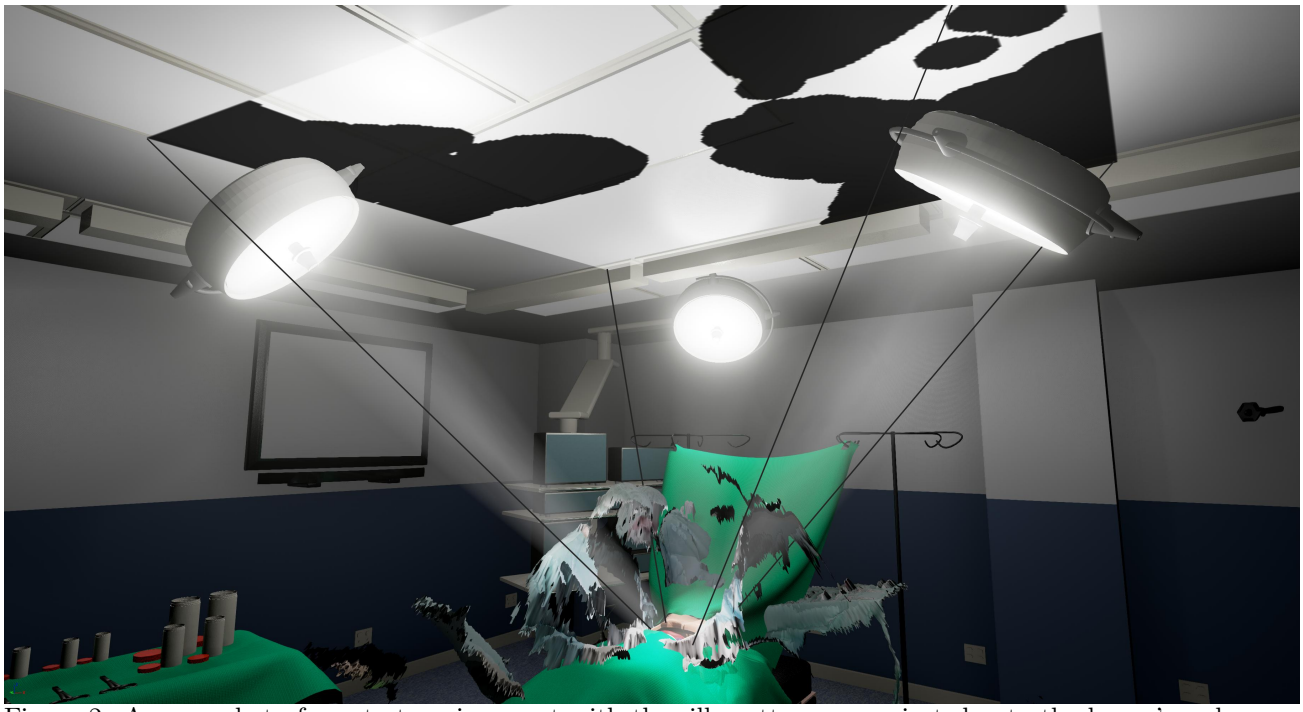

Figure 2: A screenshot of our test environment with the silhouette map projected onto the lamps' workspace. You can see the operation site in the bottom center of the image, a model of the surgeon, surgical staff and the surroundings of the operating table as it was captured by the depth camera, and the view frustum used for the rendering in black lines. This illustrates the idea of the workspace maps. The rendering of the point cloud and subsequent processing basically yields an "inverse shadowmap" from the point of view of the operation site.

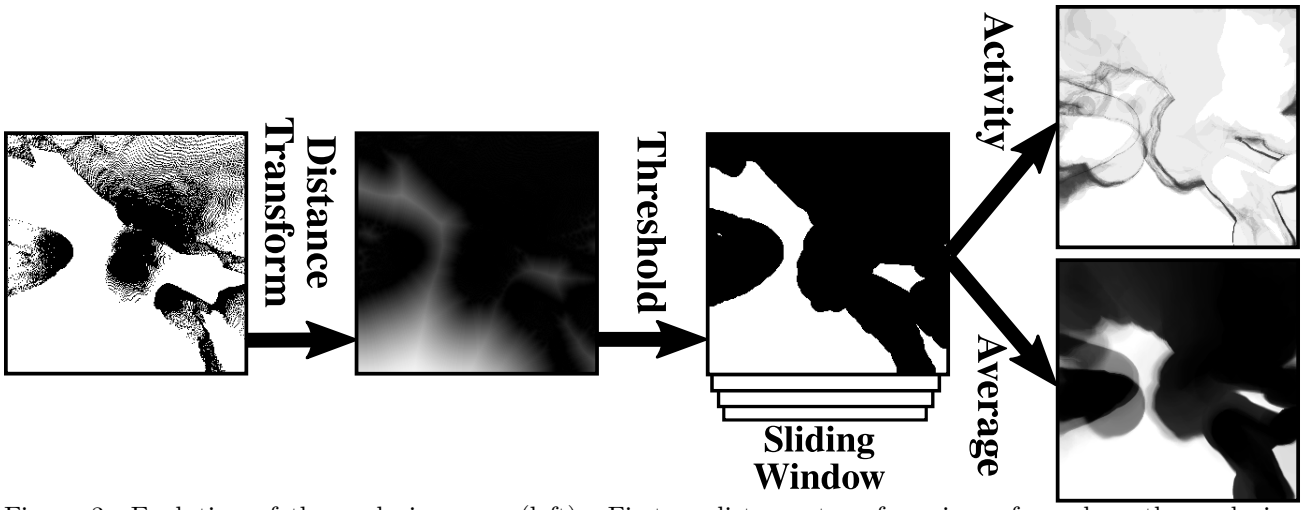

Figure 3: Evolution of the occlusion map (left). First, a distance transform is performed on the occlusion map. On the resulting distance map, a thresholding is applied to yield a map containing an approximation of the silhouettes of the surgeon and surgical staff. This silhouette map is saved in a sliding window to compute the temporal maps (on the right). The first temporal workspace map contains for each pixel the activity, i.e. how often that pixel changed from occluded to not occluded or vice versa. The other map contains the average occlusion of each pixel.

distance to the next occluder at any point. The second map approximates the silhouettes of the surgeons and surgical staff according to a pre-defined threshold. We save this so called *silhouette map* in a sliding window to create the last two maps with temporal information, namely how often every pixel was occluded over a specified amount of time and how often the occlusion status has changed over that interval, denoting how active specific areas are. We call this set of five maps *workspace maps*.

In the fourth and last stage of the pipeline, these maps are analyzed, the optimal positions are computed and the lamps are moved accordingly. Originally, we applied a simple hill climbing algorithm combined with global random sampling for this optimization. We call this the *hill-climbing optimization*. Our current work improves this step of the pipeline and compares the hill-climbing/random-sampling optimization method with the widely used particle swarm optimization, as described in the next section. The last step moves the lamps to these optimized positions while avoiding collisions between the lamps. It currently uses a simple critically damped spring-mass system to move the lamps to the destination position with a using a pre-defined maximum acceleration.

2.2 Our Novel Optimization Method

Even if our original implementation works pretty well in our test cases, it has two major drawbacks: first, the optimization tends to get stuck in local optima which result in either bad lighting conditions or an increased movement of the lamps which can be distracting for the surgeon. Second, we have chosen the optimization parameters, i.e. the weights for the movement, the occlusion, etc, experimentally by hand. Actually, these parameters depend on the type of surgery and on the surgeon's individual preferences. For instance, some surgeons feel more distracted by moving lamps and prefer to accept slightly worse lighting conditions. Obviously, determining these parameters manually for each surgeon and each kind of operation is not an option, particularly with regard to the partly counterintuitive behavior of these parameters.

In this section, we present our solution to overcome both of these drawbacks. First, we improved the runtime optimization. To do this, we applied the particle swarm optimization[4] to find a global optimum in a scalarized problem. Second, we introduce a novel meta optimization scheme that relies on the multi-objective enabled non-dominated sorting particle swarm optimizer as presented in.[5]This meta optimization basically computes the Pareto front for a pre-recorded surgery. The selection of an individual Pareto solution delivers a set of parameters that can be used for the runtime optimization. The selection is intuitive and easy and can be done

with respect to the surgeon's individual preferences. This meta optimization can be done in an offline process and it has to be performed only once for each type of surgery.

In the following we will detail the two steps of our novel optimization method.

2.2.1 Runtime-Optimization

The multi-dimensional domain space of the runtime-optimization is defined by the previously mentioned workspace maps and the distances between the lamps. We consider both the angular distance around the center point of the workspace and the euclidean distances between the lamps. The purpose of the distances is to ensure that the goal of a uniformly lit site is met by distributing the lamps as evenly across the workspace as possible and feasible. The workspace maps provide information about the valid positions on the workspace together with temporal information on the past feasibility of those positions. Based on these temporal information we can make assumptions about temporal coherence. Areas that were occluded for a long time are likely to also be occluded in the future, as well as areas with high activity in the past will probably experience high activity in the future and vice versa. Subsequently, we multiply the temporal maps and use the result as I_{temp} in the following weighted sum.

$$C(\mathcal{L}) = c_{distances} + c_{maps} \tag{1}$$
$$\text{where } \mathcal{L} = \{L_1, \ldots, L_n\} \text{ is the set of all lamps' positions.}$$

$$c_{distances} = w_{ang} \min_{\forall i,j \in [1,|\mathcal{L}|]} \sphericalangle(L_i, L_j) + w_{eucl} \min_{\forall i,j \in [1,|\mathcal{L}|]} dist(L_i, L_j)$$

$$c_{maps} = \sum_{i=1}^{|\mathcal{L}|} w_{temp} I_{temp}(L_i) + w_{occ} I_{occ}(L_i) + w_{move} dist(L_i^t, L_i^{t+1})$$

All terms of this weighted sum are normalized to range between 0 and 1, with 0 being the best possible solution and 1 the worst, before they are scaled by their respective weights. The two terms in $c_{distances}$ are the minimal angular and minimal euclidean distances between the lamps, which aims to position the lamps evenly on the workspace. c_{maps} combines the workspace maps for the individual lamps L_i. The first term considers the multiplied temporal maps at the position of the lamp ($I_{temp}(L_i)$) and the second term holds the value of the occlusion map ($I_{occ}(L_i)$). The final term in c_{maps} yields the distance between the proposed and the current lamp position in order to avoid long movements.

2.2.2 Meta-Optimization

Unfortunately, the weights of the runtime optimization behave partly counter-intuitive and finding an optimal configuration can be difficult. Even though the weights relate to simple concepts, the interplay between the different weights is non-trivial. This is a general problem of weighted global criterion methods, such as the weighted sum, as reported by numerous sources[6–8] . Fortunately, the movement in surgeries is overall very consistent and we have access to an operating room to record interventions. This enables us to use meta-optimization on the pre-recorded data to determine optimal sets of weights ($w_{ang}, w_{eucl}, w_{temp}, w_{occ}$ and w_{move}) in advance. Furthermore, the weighted sum is not strongly dependent on particular movements, but rather on the general distribution of occluders which remains fairly constant over most surgeries.

The dimensions of the search space of our meta-optimization are the weights of the runtime optimization. The dimensions of the objective space are easy-to-understand metrics for the performance of the lamps: they correspond to the distance the lamps had to move on the workspace and the estimated minimal, maximal and average occlusion of the lamps. The transformation between the spaces can be described by the function f below (Equation 2 and Figure 5). With these metrics, a surgeon can easily pick out a set of parameters that represents their preferences.

The evaluation of the lamps performance is done by recording their paths throughout the simulation as well as estimating the occlusion of the lamps by counting the pixels of the silhouette map that overlap with the projection of the lamps onto the workspace (see Figure 4).

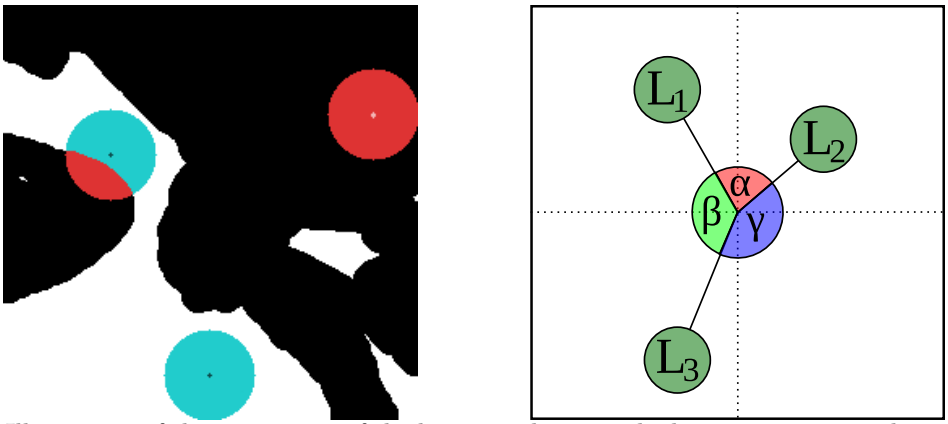

Figure 4: (left) Illustration of the estimation of the lamps occlusion. The lamps are projected onto the silhouette map to their corresponding positions and the overlapping black pixels counted. The overlapping pixels are marked red and the occlusion-free pixels are blue in this illustration. The number of red (overlapped) pixels is then divided by the number of pixels of the lamps. The corresponding occlusion value assigned to this constellation is therefore ~ 0.4 overall. (right) Illustration of the angular distance which is measured between all lamps relatively to a reference point in the center of the lamps' workspace.

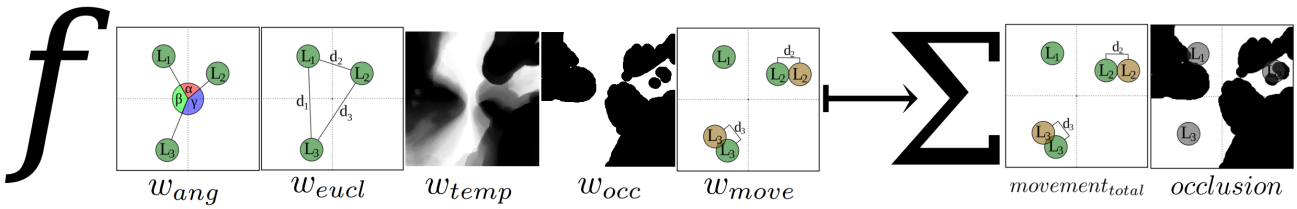

Figure 5: Visualization of the objective function of the meta-optimization. From left to right, it takes the minimal angular and minimal euclidean distances, the values of the multiplied temporal maps and the silhouette map, and the distances between the current and the proposed state. The optimization presents its performance with the total distance traveled by the lamps and the minimal, maximal and average occlusion of the lamps.

The interface for the parameter selection could be realized by a simple slider that controls the total movement of the lamps and shows the corresponding estimated occlusion based on the results of the meta-optimization. Internally, it would find the non-dominated solution vector that is closest to the chosen lamp movement and has the best values for occlusion and use the corresponding objective vector for the runtime-optimization.

$$f(w_{ang}, w_{eucl}, w_{temp}, w_{occ}, w_{move}) \rightarrow \{occlusion_{min}, occlusion_{max}, occlusion_{avg}, movement_{total}\} \quad (2)$$

3. RESULTS

We have implemented the extension of the pipeline using C++ with OpenCV[9] for computer vision tasks, just like the rest of the pipeline, with the addition of PaGmo[10] for the optimization algorithms.

We have captured an RGB-D (video and depth) recording of a six-hour open abdominal surgical procedure taken from a Microsoft Kinect v2 positioned directly above the operating table. The recording includes the surgical site, the surgical team, and the instrument trays. To facilitate progress in the field, we are making this recording available to enable other researchers without access to operating rooms to develop and validate their algorithms using real-world data.

For the evaluation, we adapted the objective function of the hill-climbing optimization so that it uses the same objective function as our novel runtime-optimization. Hence, the results are not directly comparable to those presented in our previous paper.[3]

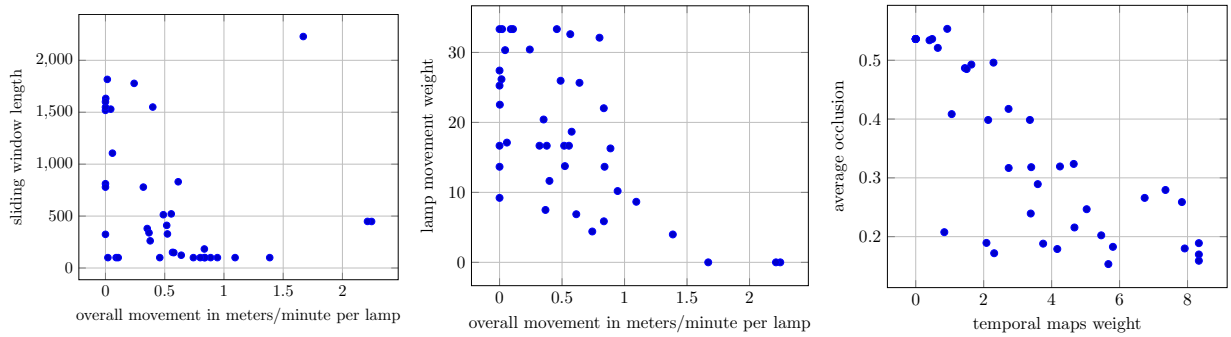

Figure 6: Scatter plots of the results of a run of the meta-optimization. The x-axes of the two left plots denote the average velocity of the three lamps in meters per minute. The y-axes show the sliding window length and the lamp movement weight (w_{move}) respectively. The outliers on the right of the leftmost plot are due to w_{temp} being 0 and consequently the sliding window not being used. The rightmost plot shows on the x-axis the temporal maps weight (w_{temp}) and the average occlusion of the lamps on the y-axis. The occlusion is computed as illustrated in Figure 4.

With this configuration, we ran the meta-optimization on a 6-minute excerpt of the recorded intervention. The result is presented in Figure 6. Each point in these plots represents the results of a simulation on this excerpt with the corresponding parameters for the runtime-optimization. As can be seen, the weights of the runtime-optimization map well on the corresponding fitness value of the meta-optimization, meaning that, for example, a higher value for the lamp movement weight w_{move} results in less movement overall (center plot). The left plot shows how retaining more information about the past, i.e. a higher value for the sliding window size, results in less movement due to the maps being more steady and long-term-viable placement being preferred. The rightmost plot shows how a bigger weight for the temporal maps (w_{temp}) results in less average occlusion. But overall, there is still much room for interdependencies between the different weights.

The plots in Figure 7 show the Pareto fronts in the objective space. The left plot shows very nicely that there is a big tradeoff to be made between less movement and better average occlusion. The Pareto front follows a $y = 1/x$-type curve, deviations are mostly due to luck or lack thereof of the optimization in finding long-term stable positions. The Pareto front in the right plot is divided into roughly two clusters, with the particles that perform the best in terms of occlusion variance also being the ones moving the least. This is due to the runtime-optimization with more movement finding optimal positions on the other side of a completely occluded area through which a lamp has to pass.

The above results of the meta-optimization allow us to easily pick respective parameters sets from the Pareto front to influence the behavior during the runtime optimization. To do that, we chose two parameter sets: 1., we used the weights that produce the smallest average occlusion and 2. we used the weights that minimize the movements of the lamps. With this configuration, we performed measurements on three different excerpts of our recording of a surgical intervention. The first excerpt is the same as the one on which we did the meta-optimization and can be seen in Figure 8. The other two excerpts were from earlier in the intervention, one from about one hour into the intervention (Figure 9) and the other one from about four hours into the intervention (Figure 10). These measurements show that the optimization remains fairly stable over different parts of the same intervention. The difference between the new and the old optimization mostly stems from the global optimization of the particle swarm optimization which leads to further optima being discovered, leading to slightly more movement but also less occlusion if allowed.

Our results show, that even with a relatively short training period, the trained values result in similar lamp behavior in different situations during the intervention.

The performance of the new runtime-optimization actually shows the wanted behavior of moving the lamps to good positions, even if they are further away but then remaining there for longer stretches of time. This can be clearly seen in Figure~\ref{fig:MovementComp}, which shows the movement of the lamps with the movement optimized version of the optimizations of Figure 9.

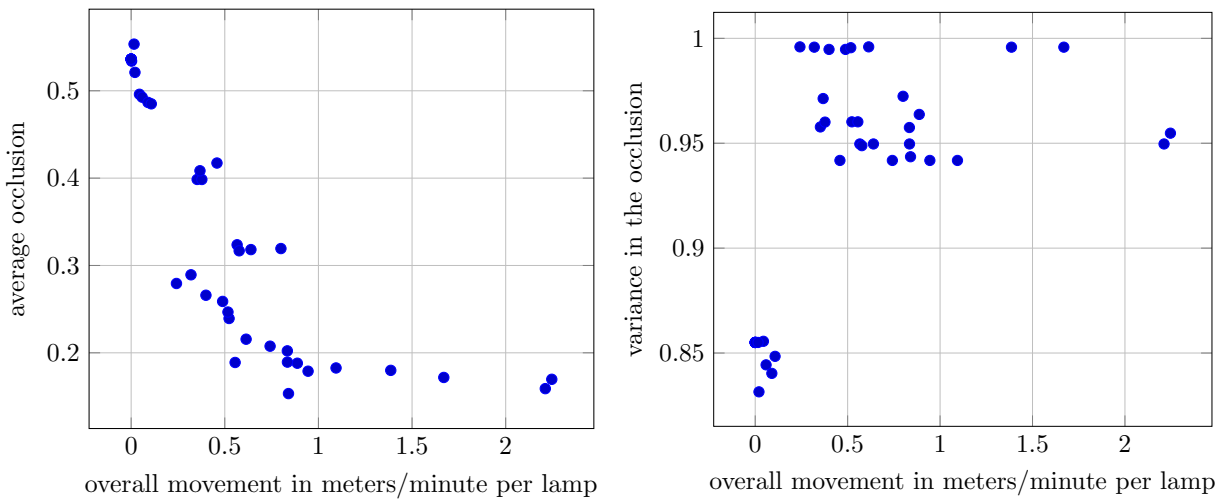

Figure 7: Plots of the Pareto fronts of the meta-optimization. The x-axes in both plots, again, denote the average velocity of the three lamps in meters per minute. The y-axis of the leftmost plot is the average occlusion over the training period. The y-axis of the rightmost plot shows the variance of the occlusion as $occlusion_{max} - occlusion_{min}$.

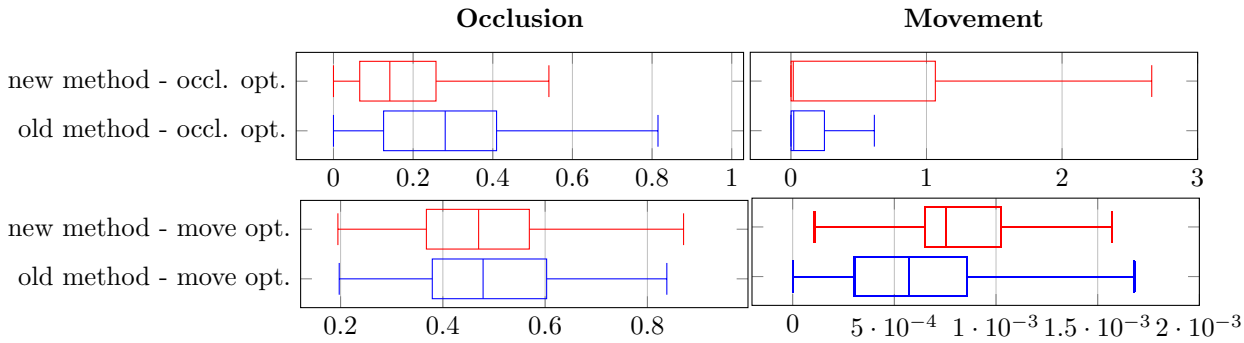

Figure 8: Box plots showing the relative occlusion and movement of all lamps with the chosen movement and occlusion optimized weights with the old hill-climbing optimization and the new runtime-optimization. The x-axes are the relative occlusion and the velocity of all lamps in meters per minute. The excerpt of the recording used for these plots is the one on which the meta-optimization was performed. The runtime-optimization outperforms the hill-climbing optimization in occlusion but moves further while doing so. Please note the different scales of the x-axes. The movement optimized weights lead to almost no movement overall, still, the new method accomplishes to get far less occlusion.

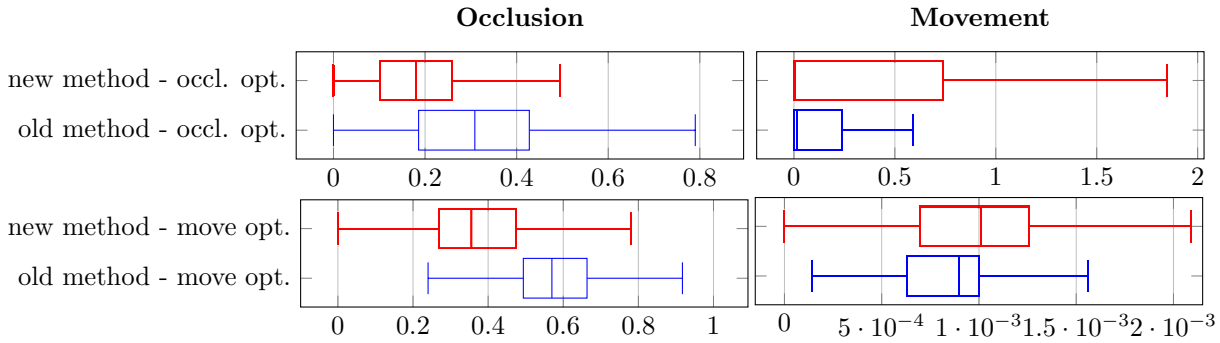

Figure 9: The same box plots as above, but with data obtained by running the simulation on another excerpt of the intervention several hours earlier in the intervention. As you can see, the overall results are very similar to the ones shown above. The only notable difference is that the new runtime-optimization with move optimized weights produces a lot less occlusion with only slightly more movement. The performance in occlusion is actually similar to the old hill-climbing optimization with occlusion optimized weights.

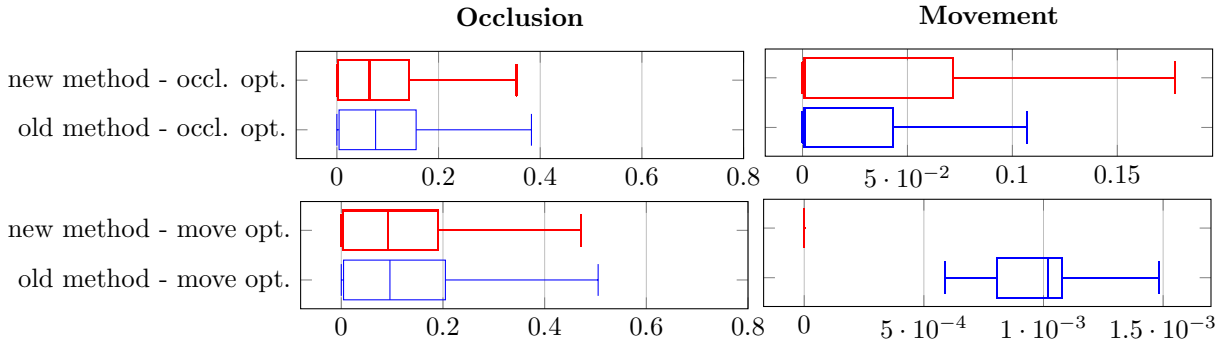

Figure 10: The same box plots again, this time with data obtained by running the simulation on another 15 minutes excerpt from just an hour earlier than the training excerpt in the same intervention. This part of the intervention was particularly easy for the optimization. Even the occlusion optimized weights produced very little movement while the new optimization with movement optimized weights didn't move at all.

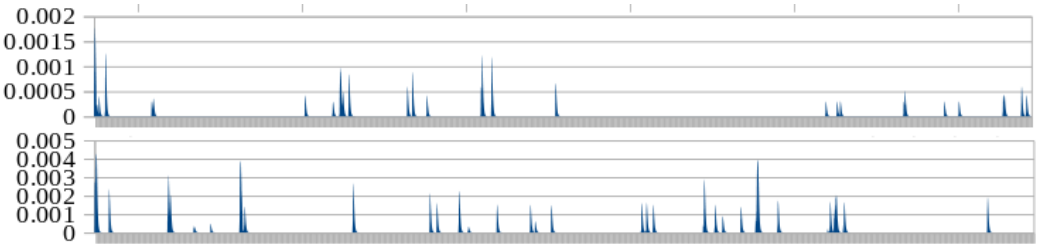

Figure 11: Plot of the movement of the lamps at the simulations corresponding to Figure 9. The top plot shows the movement of the new runtime optimization and the bottom plot the old hill-climbing optimization. The x-axis is the run-time of the simulation which is about 15 minutes. It shows clearly, that the runtime optimization leads to longer stretches of no movement when compared to the old method.

4. CONCLUSION

We have presented a novel two-tiered approach to optimize the movement of autonomous surgical lamps. Our method includes a runtime optimization as well as a meta optimization. Our meta-optimization delivers a simple set of easy to understand parameters that the surgeon can choose from, depending on their preference. Moreover, the meta-optimization shows that there is a clear tradeoff between less movement and less occlusion. Optimizing one will inevitably lead to worse outcomes regarding the other. This is how our meta-optimization can help surgeons in picking the right parameters as the results also show that our meta optimization delivers much better results than the hand selected parameters we used before.

However, there is still room for future improvements. For instance, we want to evaluate how well the weights calculated by the meta-optimization are applicable to different scenarios in the operating room than the one analyzed. We believe the runtime-optimization should be fairly independent of the situation in the operating room, but we need to investigate further before making any definite claims.

In addition there are more parameters for the optimization the impact of which we need to study further. These are parameters like the length of the sliding window for the temporal workspace maps and the function with which the individual frames are weighted based on the time they were taken, which could impact the optimal values for the weight for the temporal maps. Another parameter would be the maximum velocity and acceleration of the lamps. The above measurements were taken with fairly fast lamps which leads to relatively high movement. Lower values should lead to less movement while the impact on the occlusion needs to be studied further.

In the future we want to explore several different paths to improve the pipeline and optimization as well as our verification method. For example, we want to evaluate the value of more information about the environment around the site. For that purpose, we are currently working on creating a completely simulated surgical procedure with motion tracking and 3D modelling techniques. In this simulated surgery, we can place virtual depth-cameras to feed into our pipeline, as well as accurately calculate how much light from the lamps reaches the site and it can be used to perform user studies.

Furthermore, we are working on simulating the light-beam-geometry of typical surgical lamps with computer graphics techniques. This would allow us to estimate much better how much of the lamps light reaches the site. Together with a reconstruction of the geometry of the operating site we could also evaluate the lighting situation in cavities in the site.

With a combination of site- and hand-detection and -tracking, we would like to determine the area and direction in which the surgeon is currently working. With this information, we would try to maximize the light focused on the spot the surgeon is operating on and also the amount of light ending up in the surgeons eyes. This might be done by changing the center around which the lamps should loosely be positioned.

We would also like to generalize our algorithms to optimize other devices that require line of sight, such as optical navigation systems or computer monitors that the surgeon must view.

Last but not least, we are preparing a user study with real surgeons to evaluate the autonomous lamps. For this we created a virtual reality environment in which the surgeon needs to perform a familiar task in the familiar environment of a virtual operating room while the lamps can either be moved manually by the surgeon or autonomously by our algorithm.

ACKNOWLEDGMENTS

We would like to thank the Asklepios Klinik Barmbek and the International Neuroscience Institute for their support. This work was partially supported by the grand "Creative Unit – Intra-Operative Information".

REFERENCES

[1] International Electrotechnical Commission et al., "Particular requirements for the basic safety and essential performance of surgical luminaires and luminaires for diagnosis," tech. rep., IEC 60601-2-41:2009+AMD1:2013 (2013).

[2] Knulst, A. J., Mooijweer, R., Jansen, F. W., Stassen, L. P., and Dankelman, J., "Indicating shortcomings in surgical lighting systems," *Minimally Invasive Therapy & Allied Technologies* **20**(5), 267–275 (2011).

[3] Teuber, J., Weller, R., Kikinis, R., Oldhafer, K.-J., Lipp, M. J., and Zachmann, G., "Autonomous surgical lamps," in [*Jahrestagung der Deutschen Gesellschaft für Computer- und Roboterassistierte Chirurgie (CURAC)*], (Sept. 2015).

[4] Kennedy, J., "Particle swarm optimization," in [*Encyclopedia of machine learning*], 760–766, Springer (2011).

[5] Li, X., "A non-dominated sorting particle swarm optimizer for multiobjective optimization," in [*Genetic and Evolutionary Computation Conference*], 37–48, Springer (2003).

[6] Hwang, C.-L. and Yoon, K., [*Multiple attribute decision making: methods and applications a state-of-the-art survey*], vol. 186, Springer Science & Business Media (2012).

[7] Hobbs, B. F., "A comparison of weighting methods in power plant siting," *Decision Sciences* **11**(4), 725–737 (1980).

[8] Marler, R. T. and Arora, J. S., "Survey of multi-objective optimization methods for engineering," *Structural and multidisciplinary optimization* **26**(6), 369–395 (2004).

[9] Bradski, G. et al., "The opencv library," *Doctor Dobbs Journal* **25**(11), 120–126 (2000).

[10] Biscani, F., Izzo, D., and Yam, C. H., "A global optimisation toolbox for massively parallel engineering optimisation," *arXiv preprint arXiv:1004.3824* (2010).