A Comparative Study of Matrix Completion and Recovery Techniques for Human Pose Estimation

Dennis Bautembach
Institute of Computer Science, FORTH
Computer Science Department, University of Crete
denniskb@ics.forth.gr

Iason Oikonomidis
Institute of Computer Science, FORTH
oikonom@ics.forth.gr

Antonis Argyros
Computer Science Department, University of Crete
Institute of Computer Science, FORTH
argyros@ics.forth.gr

ABSTRACT

We present a comparative study of three matrix completion and recovery techniques, applied to the problem of human pose estimation. Human pose estimation algorithms may exhibit estimation noise or may completely fail to provide estimates for some joints. A post-process is often employed to recover the missing joints' locations from the available ones, typically by enforcing kinematic constraints or by using a prior learned from a database of natural poses. Matrix completion and recovery techniques fall into the latter category and operate by filling-in missing entries of a matrix, with the available/non-missing entries being potentially corrupted by noise. We compare the performance of three such techniques in terms of the estimation error of their output as well as their runtime under varying parameters. We conclude by recommending use cases for each of the compared techniques.

KEYWORDS

Matrix completion, matrix recovery, human pose estimation, comparative study

ACM Reference Format:
Dennis Bautembach, Iason Oikonomidis, and Antonis Argyros. 2018. A Comparative Study of Matrix Completion and Recovery Techniques for Human Pose Estimation. In *PETRA '18: The 11th PErvasive Technologies Related to Assistive Environments Conference, June 26–29, 2018, Corfu, Greece.* ACM, New York, NY, USA, 8 pages. https://doi.org/10.1145/3197768.3197791

1 INTRODUCTION

The estimation of human motion from visual input comprises a central category of problems in the field of computer vision. Many problems are defined within this category, including the estimation and tracking of human body pose [24] and the estimation and tracking of human hand pose [9]. Collectively, we refer to these problems as human motion estimation. This area of research is very active since at least the early 80s. It has received renewed interest with the introduction of depth sensors [34], the success of deep learning for computer vision tasks [17], as well as with the recent increase of interest in Augmented and Virtual Reality (AR/VR) applications. Unobtrusive capturing and monitoring of human motion is a core component of many applications including natural user interfaces, AR/VR applications, medical assessment and rehabilitation and more. Assistive environments can incorporate such a component, enabling natural interaction between the assistive system and the assisted person. Consider as a concrete and motivating example, the line of work on human pose estimation by Michel et al. [22, 23]. This algorithm operates on visual input provided by an RGBD camera and outputs 3D information for the human body joints that exceed some minimum estimation confidence. However, for building applications like vision-guided personal fitness trainers [11] or for supporting clinical applications of smart walkers [32], it is very important that reasonable estimates of the positions of missing joints are available.

Both human body pose estimation and human hand pose estimation exhibit several difficulties. These include sensor noise, the high number of Degrees of Freedom (DOF) of the human hand and body, the high versatility and large range of human motion and the inevitable occlusions (self-occlusions or occlusions from the environment). Because of occlusions, it is common to have poorly observed or unobserved parts of the target, in turn leading to inaccurate or totally missing estimations regarding these parts. There are several ways to alleviate this problem. Many methods employ a post-processing step to estimate or refine uncertain or missing joints. They do so by enforcing constraints that can be derived either from the kinematic chain of the observed body, or induced from datasets that contain natural poses of the target.

A common category of techniques to accomplish this goal is matrix completion [4, 39] and recovery [19]. Matrix completion is the task of completing missing values of a matrix, usually under the assumption that the rank of the resulting matrix is minimized, essentially enforcing linear dependency of the entries. Matrix recovery works similarly to this, under the additional assumption that the known values are contaminated with noise. In this case, the whole matrix is recomputed, or recovered, including both the missing and the observed values. Applied to the problems of human motion estimation, these approaches provide a non-parametric way to model the prior over natural poses. The only requirement is a dataset of comparable poses, which, together with the current pose, serves as the to-be-completed input matrix. At runtime, an estimated pose with uncertain or missing entries can be post-processed using these techniques to yield a pose that resembles the pre-acquired poses.

In this work we compare three different techniques for matrix completion and recovery [4, 19, 39] for the task of recovering the positions of missing joints given an estimate of a human body pose.

PETRA '18, June 26–29, 2018, Corfu, Greece

ACM ISBN 978-1-4503-6390-7/18/06…$15.00
https://doi.org/10.1145/3197768.3197791

We focus on these techniques among other alternatives for pose completion because they are easy to implement and require only a moderately sized pose dataset (a few thousand poses). An additional advantage is that these approaches can be used to target a limited pose space in a straightforward manner by appropriately limiting the employed dataset.

The suitability of these techniques, e.g., for Human-Computer Interaction (HCI), depends on the requirements of the target application. Specific parameters include the error tolerance and the execution time of the complete pipeline. Therefore, in order to assess the suitability of the selected techniques for specific applications, we experiment with several parameters of our basic setup, including dataset size, number of missing joints, and the effect of observation noise.

In the next section we present a short overview of works on human motion estimation, with the common denominator that the problem of grossly corrupted estimations receives special treatment, either in post-processing, or as a separate methodological element that must be integrated in the method. In Section 3 we present a brief overview of the three compared methods, as well as of the methodological elements of the comparison. These include details on the dataset we use in our experimental study as well as the basic design of the experiments. In Section 4 we present several experiments we conducted, discussing the results we obtained. We conclude this work with a brief summary of the key points.

2 LITERATURE OVERVIEW

The problem of human body motion estimation from visual input is long-standing and well studied [1, 5, 24, 37]. Similar progress has been achieved in the related problem of human hand motion estimation [9, 27, 41, 44]. The term "visual input" refers to any passive or minimally intrusive observation modality. Specifically, this includes regular RGB images acquired using monocular, stereo, or multi-camera configurations. This also includes depth sensors that can either operate passively, for example using stereo reconstruction, or actively, emitting infrared light in the scene to estimate depth [34]. This definition does not include observation modalities that require specialized markers [30, 46] or other ways of rigging the observed scene [47].

Both problems exhibit difficulties such as the number of DOFs of the target, the versatility and large range of motion, sensor noise, and occlusions that may occur either because of other objects in the environment or due to the tracked object itself (self-occlusions). Despite these shared difficulties, most of the methods in the related literature tackle tracking human bodies and human hands separately, mostly due to the scale difference of the targets. Nevertheless, lately there has been some effort for unified body and hand pose estimation from a single system [15, 36].

The problem of recovering a plausible pose given a noisy estimation with potentially missing entries is central to all methods that perform human motion estimation. Matrix completion and recovery techniques can be used to tackle this problem. Alternative approaches are also explored in the related literature, including imposing assumptions regarding the motion in consecutive frames [42] and modeling the space of natural poses. This last approach can be adopted either implicitly, especially in learning frameworks [28], or explicitly [6] as a post-processing step to refine the estimated pose.

Completing and correcting estimated poses is a major challenge for all techniques that perform human motion estimation. Various approaches are adopted towards this goal. Indicatively, several methods [16, 21, 45] employ physics simulation to enforce physical plausibility in the estimated hand poses. Another approach is to use inverse kinematics [10, 44], essentially enforcing kinematics constraints on the estimated poses. In a different approach, methods [40, 41] hierarchically regress the pose of parts of the hand without globally enforcing pose constraints, potentially leading to implausible poses. A learned pose prior is implicitly enforced by Douvantzis et al. [7] in the form of a standard dimensionality reduction technique. A similar approach is also adopted by Oberweger et al. in their line of work [27, 28], by incorporating a low-dimension layer (called a 'bottleneck') in the last layer of the learned network. A post-processing step is employed by Ciotti et al. [6] that refines the estimated hand pose using the occlusion cue as a measure of uncertainty. Roditakis et al. [35] estimates hand pose during hand-object interaction by considering spatial constraints induced by the observed hand-object contact points. Deep-learning based methods [3, 13, 14, 18, 33, 38] for human pose estimation use large datasets to learn the space of natural human poses. On top of this, Brau et al. [3] enforces constraints on body part lengths. Furthermore, a few works [26, 42] use large training sets and architectures similar to that of Oberweger et al. [28], incorporating bottleneck layers. Baak et al. [2] propose performing a lookup for the most fitting pose in a large database of candidate poses. Several works [8, 20, 25, 49] use human kinematics and physically plausible joint limits to recover natural human poses. Tekin et al. [43] exploit spatio-temporal information on large training sets. Yu et al. [48] jointly estimate shape and pose, essentially imposing observed shape constraints.

Matrix completion and recovery [4, 19, 39] can provide a viable option for such approaches, implicitly modeling the space of natural poses requiring only a dataset of natural poses of the target, and adding a potentially lightweight post-processing step to the computational pipeline. The present work serves as a comprehensive study of strengths and weaknesses of each compared approach. Our hope is that this work can prove useful in improving the results of methods that naturally yield uncertain and/or grossly corrupted estimations.

3 DESIGN OF COMPARATIVE STUDY

In Section 3.1 we provide a brief overview of the three compared methods for matrix completion and recovery. All approaches aim to minimize the rank of the computed matrix, but since this is an NP-hard problem, the methods adopt approximations of it. In Section 3.2 we describe the general methodology we followed to obtain the experimental results presented in Section 4.

3.1 Matrix Completion and Recovery

Inversion-based Matrix Completion (IBMC): The first approach is termed *IBMC* after the core arithmetic operation that is used to complete the missing values, that is, a multiplication by a pseudo-inverse matrix. The input to the method is a matrix that has a

missing block. Following the description provided in the works by Sinha et al. [39] and Owen et al. [31], this approach starts by having the missing values rearranged to the bottom-right of the matrix X, in a block-matrix p_2:

$$X = \left[\begin{array}{c|c} D_1 & p_1 \\ \hline d_2 & p_2 \end{array}\right]. \tag{1}$$

Under the assumption that both matrices X and D_1 have the same rank k, it is possible to show that the missing values p_2 can be expressed as

$$p_2 = d_2 \cdot (D_1)^+ \cdot p_1. \tag{2}$$

In practice, this assumption implies that the matrix X has linearly dependent entries that can therefore be exactly recovered.

Gradient Descent Matrix Completion (GDMC): The second approach, called *GDMC*, is an implementation of the method proposed by Candès et al. [4]. The main goal is to complete the missing values so that the rank of the resulting matrix is minimized. Following the notation in that work [4], we denote the matrix including the missing values as M, with X denoting only the observed ones. The minimization problem can then be formulated as

$$\begin{array}{ll} \text{minimize} & rank(X) \\ \text{subject to} & X_{i,j} = M_{i,j} \qquad (i,j) \in \Omega, \end{array} \tag{3}$$

where $rank(X)$ is defined to be equal to the rank of the matrix X and Ω indexes the observed values. After observing that this optimization problem is NP-hard, and that all known algorithms that solve it have doubly-exponential complexity, the authors proceed to approximate it. They select the sum of singular values as an approximation to the rank of a matrix, and formulate the resulting minimization problem. Following the same notation and defining the nuclear norm as $\|X\|_* = \sum_{k=1}^{n} \sigma_k(X)$, where $\sigma_k(X)$ denotes the k-th largest eigenvalue, this optimization problem can be formulated as:

$$\begin{array}{ll} \text{minimize} & \|X\|_* \\ \text{subject to} & X_{i,j} = M_{i,j} \qquad (i,j) \in \Omega. \end{array} \tag{4}$$

This minimization problem can be solved efficiently using gradient descent, since the resulting objective function is convex.

Matrix Recovery with Lagrange Multipliers (MRLM): The work by Lin et al. [19] tackles the similar problem of matrix recovery. Additionally to completing missing values of the input matrix, matrix recovery also re-estimates the provided values. This is done under the assumption that the observed values are contaminated with noise. Therefore, the low-rank assumption that is used to complete the missing values can also help in the task of removing the noise in the observed ones. Lin et al. start by formulating Principal Component Analysis (PCA) as the problem of computing a low-rank matrix that has entries close to the input matrix:

$$\begin{array}{ll} \text{minimize} & \|E\|_F \\ \text{subject to} & rank(A) \leq r, M = A + E, \end{array} \tag{5}$$

where again, M is the input matrix, $\|\cdot\|_F$ denotes the Frobenius norm, and r is the target matrix rank. This problem can be efficiently solved using the Singular Value Decomposition (SVD) of M, and by keeping only the r largest singular values. Then the more general problem is treated, in which the input matrix M, apart from noisy values is also assumed to have grossly corrupted or missing values. This leads to the following problem formulation

$$\begin{array}{ll} \text{minimize} & \|A\|_* + \lambda\|E\|_1 \\ \text{subject to} & M = A + E, \end{array} \tag{6}$$

where $\|\cdot\|_*$ denotes the nuclear norm of a matrix, i.e. the sum of its singular values, and $\|\cdot\|_1$ denotes the sum of absolute values of matrix entries. Using this formulation, it is shown that by applying the method of Augmented Lagrange multipliers, an iterative approximation scheme converges to the exact solution in a few iterations.

The rationale behind both GDMC and MRLM is that the most plausible completion of a matrix is one that minimizes its rank. Note, however that both algorithms do not solve this problem, but rather an approximation of it. Even if they did solve the original problem, the solution would not necessarily coincide with the correct pose in the context of human motion estimation. GDMC completes a matrix without modifying it while MRLM returns a *similar* matrix with completed entries and approximately minimized rank. This additional relaxation allows it to be more robust against noise. IBMC, on the other hand, provides a closed form solution which requires the data to have a special structure, one that allows to group all missing entries of a matrix in a rectangular region.

3.2 Conducting the Experiments

We apply matrix completion to human motion estimation by appending a pose vector with missing joint locations to a database (a matrix) of known, representative poses. This "pose matrix" is then completed by one of the compared approaches, thus yielding estimates for the locations of missing joints.

Our experiments are based on the MHAD dataset [29]. MHAD captures 12 subjects performing 11 actions for 5 repetitions (thus yielding 660 distinct sequences) using multiple sensor modalities. We use the skeletal data captured using optical motion capture. First, we transform MHAD's relative rotations representation into an absolute (3D) locations representation, which we then normalize by removing global translation and rotation from the root joint. In a practical tracking scenario, where global transformation information is not provided, this normalization can be achieved by first aligning all poses in the database with the one to be completed (for example using [12]), adding only negligible overhead to the entire process. We chose an absolute locations representation because it enables inferring dependencies between joints. In a relative rotations representation on the other hand, knowing the orientation of $N-1$ joints does not convey any information about the $N-th$ joint as for example in the case of a person sitting on a chair and waving their hand. The resulting skeleton consists of 24 joints.

MHAD's 660 sequences yield in excess of one million *numerically* distinct poses, but much closer to ~50.000 *anatomically* distinct poses, roughly corresponding to the first ~500 frames of each sequence without repetitions. We restrict our experiments to this set of 50.000 poses. More specifically, the first 500 frames of the first repetition of each sequence form our test set, while the first 500 frames of the second repetition of each sequence form our pose database. Splitting the dataset along repetitions like this ensures

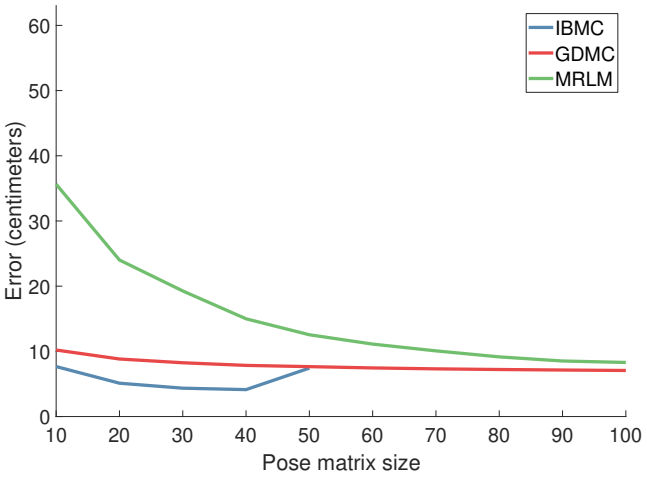

Figure 1: Error as a function of the pose matrix size. We vary the size N of the pose matrix (which consists of the N closest poses selected from the pose database) and measure the resulting average error per joint, per completed pose. While GDMC and MRLM continue to improve with more evidence, IBMC exhibits a basin and becomes unstable after ~50 poses.

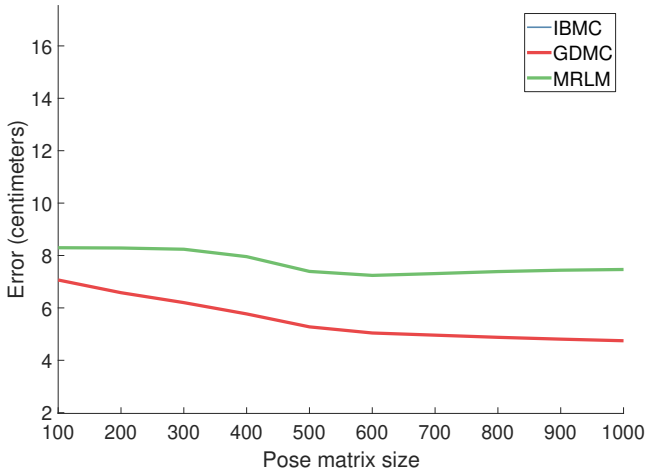

Figure 2: Same as Figure 1 for $N = 100$ to 1000. GDMC and MRLM level off at 500–600 poses.

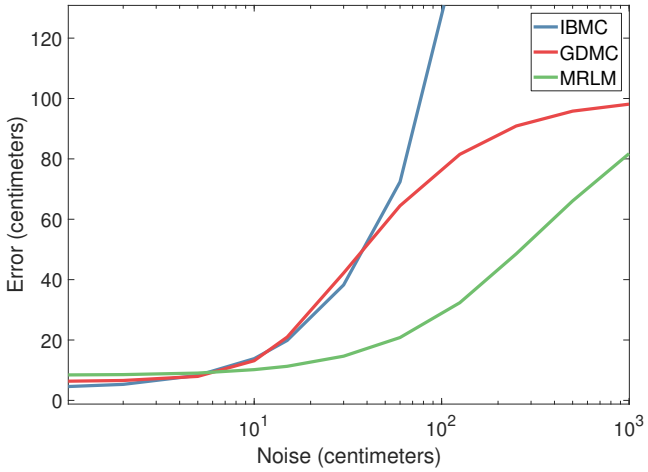

Figure 3: Error as a function of noise. We leave the pose database (and matrix) intact, but contaminate the to-be-completed pose vector with Gaussian noise, simulating uncertainty of real world sensors, and measure the resulting error. The x-axis shows noise standard deviation in centimeters on a logarithmic scale. Extreme noise values have been considered to showcase the performance of the algorithms in the broadest possible spectrum of noise contaminations.

that the pose we are trying to complete is not itself contained in the database.

For each experiment we complete 2.000 poses from the test set uniformly and at random and report the average error per pose per joint. For each pose, a "missing" joint is randomly selected and its location estimated with each of the three algorithms compared in this study. We do not use the entire pose database as the pose matrix, but only the N closest poses (under Euclidean distance) to the one we are trying to complete. This parameter affects the accuracy of the recovered pose. As we will see, all algorithms converge for N larger than about a hundred poses. This also significantly speeds up computations. As our error metric we use the Euclidean distance (in centimeters) between the estimated and the true location of a joint, averaged across all completed poses. The same error metric is used to select the N closest poses from the pose database for the pose matrix. Unless noted otherwise, we only estimate a single joint per pose. For each experiment, all the remaining parameters are set to values known to produce the best results as determined by other experiments.

4 EXPERIMENTAL RESULTS

We conducted several experiments, comparing the three matrix completion and recovery techniques with respect to their pose estimation error and their runtime. In all error plots we report the average distance in cm (in 3D space) per missing joint per frame. Thus, an error of 10cm implies that on average each estimated joint was 10cm off its true, ground truth 3D location.

4.1 Human Pose Estimation Error

Estimation error as a function of the pose matrix size: We vary the size N of the pose matrix (which consists of the N closest poses selected from the pose database) and observe the resulting error. Figure 1 illustrates the obtained results. Both GDMC and MRLM continue to improve with more evidence up to 500–600 poses at which point they level off (Figure 2). IBMC on the other hand exhibits a basin centered at ~35 poses and becomes numerically unstable after 50 poses. This happens to coincide with the estimated rank of the pose matrix in this experiment, meaning that the pseudo-inverse of a non-invertible matrix succeeds but does not produce numerically sensible/reliable results.

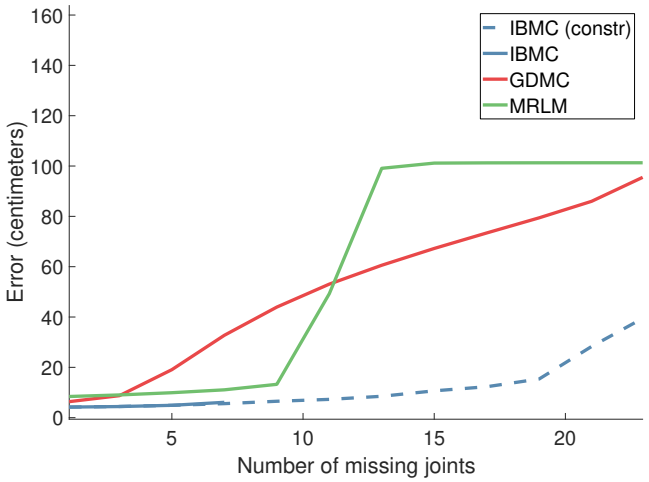

Figure 4: Error as a function of the number of missing joints.

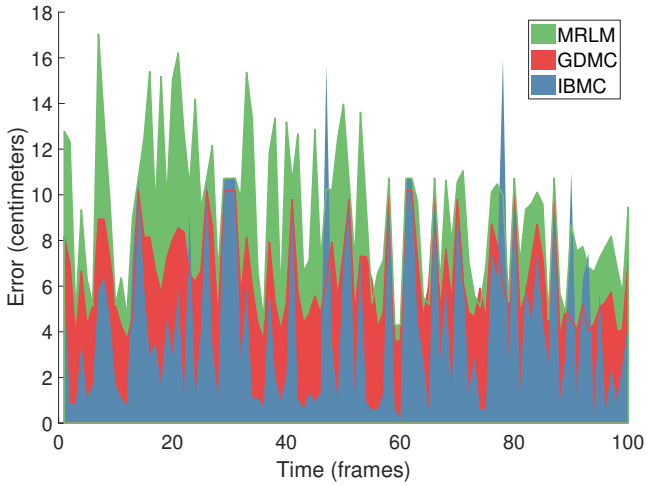

Figure 5: Error over time. We visualize the maximum error per pose produced by each of the three algorithms over a sequence of 100 temporally continuous frames. We observe a very uneven distribution of the error in the time domain, resulting in jitter in rendered sequences.

Estimation error as a function of noise: We maintain the pose database (as well as the pose matrix) intact, but contaminate the to-be-completed pose vector with Gaussian noise, and observe the resulting error. Figure 3 illustrates the obtained results. IBMC's error grows linearly, which is expected (note that the x-axis has logarithmic scale). This is because the noisy pose vector is multiplied with the pseudo-inverse of the pose matrix, thus contaminating the final result as well. The error of both GDMC and MRLM initially grows but then levels off as we approach the extreme case of a basically random pose vector. In the absence of information, both techniques degenerate to computing *some* pose that is in line with the rest of the pose matrix, and which therefore can only be so far away from any naturally occurring human pose.

Error as a function of the number of missing joints: Figure 4 shows the estimation error as a function of the number of missing joints. For up to seven missing joints IBMC performs well as the known values are characteristic enough to recover the pose. However, it becomes unstable as the size of the pose matrix surpasses its rank. We can alleviate this numerical instability by limiting the size of the pose matrix (rather than using a fixed size of 35), which produces the dashed plot in Figure 4. The qualitative characteristics of the behavior of GDMC and MRLM are similar to those of the experiment with noise. In a sense, a large number of missing joints and the presence of considerable amounts of noise can be considered as two different forms of very high uncertainty.

Estimation error as a function of time: Figure 5 shows the maximum error per pose over a sequence of 100 temporally continuous frames for each of the three techniques. We observe a very uneven distribution of the error in the time domain: long periods (~5 frames) of very low error are interrupted by abrupt error spikes. This illustrates that no technique is able to guarantee upper error bounds, but only average performance. This is also manifested as very noticeable visual "popping" when rendering completed sequences. One way to alleviate this problem would be to apply temporal smoothing as a final post-processing step.

Estimation error per joint: We visualize (Figure 6) the estimation error of each algorithm per joint. The errors are drawn from small to large, front to back. We see that the error is distributed evenly across all joints, except for IBMC, where joints higher up the kinematic chain (for example hip or shoulder) seem to be harder to estimate than end effectors (for example hands or feet).

4.2 Runtime

We vary the pose matrix size and observe the average time it takes (in seconds) *to complete a single pose* (Figure 7). The time to construct the pose matrix is not included. Our implementations are based on Matlab and thus the reported runtimes cannot serve as lower bounds for real-world applications. They do, however, allow us to compare the algorithms with one another and to determine their complexity. IBMC runs in realtime even in Matlab as its most expensive operation is the calculation of a single pseudo-inverse of a 35×72 matrix (35 poses $\times$ 24 joints $\times$ 3 coordinates). For practical purposes, GDMC is an order of magnitude slower than IBMC, and MRLM is yet another order of magnitude slower than GDMC. For completeness, and although this involves pose matrix sizes that are not really encountered in our problem, in Figure 8 we show that the execution time of GDMC grows quadratically, while for IBMC and MRLM it grows linearly.

5 CONCLUSIONS

A comparative study of three matrix completion techniques applied to the problem of human pose estimation was presented. Several experiments exposed the differences between the approaches with respect to estimation accuracy and runtime.

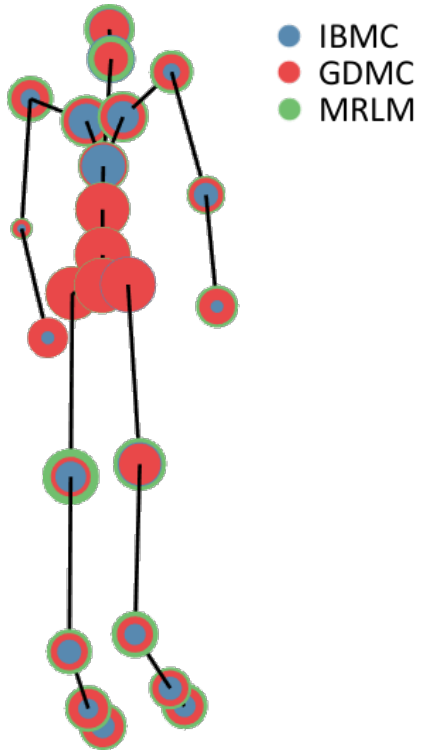

Figure 6: Error per joint. For each joint we visualize a disk whose radius depends on the average expected error in estimating this joint by each of the three techniques. The skeleton belongs to a 180cm tall man. The error disks are drawn up to this scale.

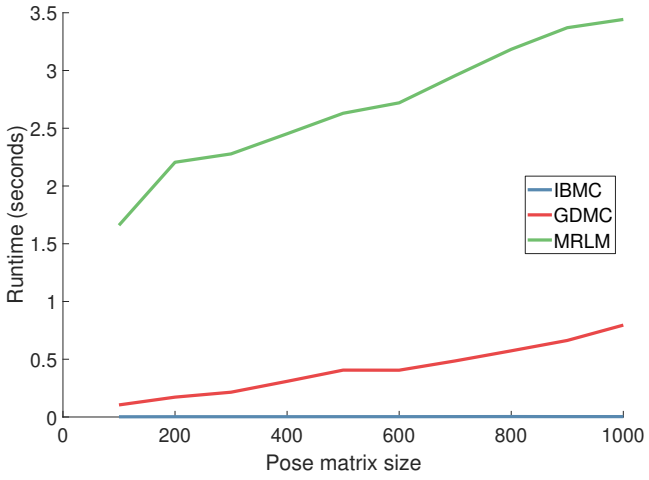

Figure 7: Runtime as a function of the size of the pose matrix. The y-axis shows the average time in seconds it takes to complete a single pose. Time for constructing the pose matrix is not included.

The parameters that affect the selection of appropriate techniques are the runtime requirements, the amount of noise present in the available joints before completion and the desired level of overall error/accuracy after completion. For most applications (low number of missing joints, low amount of noise) IBMC seems to be the best choice due to its low error, realtime performance, and ease of implementation. It does suffer from numerical instability under certain conditions though. GDMC and MRLM represent more stable alternatives where MRLM is typically more robust against noise at the cost of runtime. All algorithms produce errors of high variability and of uneven distribution in the time domain. This imposes the requirement of an extra post-processing step that smooths-out the results for interactive applications.

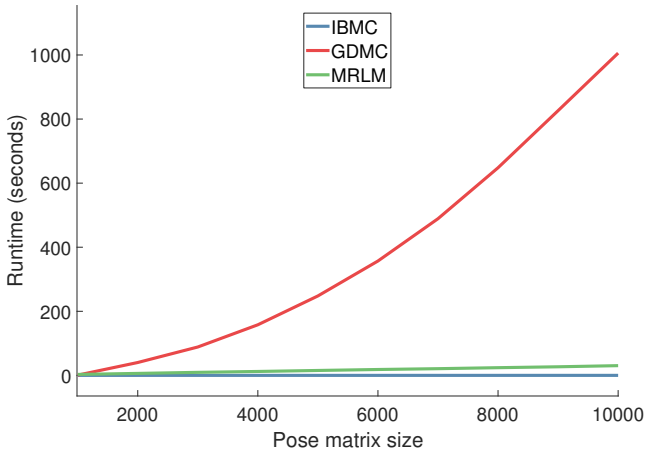

Figure 8: Same as Figure 7 but for pose matrices of sizes in the range 1.000 to 10.000, so as to illustrate the quadratic complexity of GDMC.

The different characteristics of the motion completion/recovery algorithms can guide the selection of the best choice, also in connection to the requirements of specific application domains. As an example, completion may be needed to feed online action recognition, which requires fast computation of coarse motion information. In such a setting, IBMC appears to be the best choice. On the other hand, one might think of motion recovery to support offline motion retargeting. In this context, the target accuracy is high while the runtime is not an issue and, therefore, the best choice is MRLM, especially in the case of noisy input.

Ongoing research will consider the performance of the compared algorithms as a means of completing/recovering the missing joints of specific human pose estimation algorithms such as the line of work of Michel et al. [22, 23].

ACKNOWLEDGEMENTS

This work was partially supported by the EU projects Co4Robots and ACANTO. We thank Aggeliki Tsoli for her insightful feedback and fruitful discussions, and Giorgos Karvounas for help with conducting the experiments and preparing the results. Aggeliki Tsoli and Giorgos Karvounas are members of the Computational Vision and Robotics laboratory of ICS-FORTH.

REFERENCES

[1] Mykhaylo Andriluka, Leonid Pishchulin, Peter Gehler, and Bernt Schiele. 2014. 2D Human Pose Estimation: New Benchmark and State of the Art Analysis. In *Computer Vision and Pattern Recognition*. 3686–3693. https://doi.org/10.1109/CVPR.2014.471

[2] Andreas Baak, Meinard Muller, Gaurav Bharaj, Hans Peter Seidel, and Christian Theobalt. 2011. A data-driven approach for real-time full body pose reconstruction from a depth camera. *International Conference on Computer Vision* (2011), 1092–1099. https://doi.org/10.1109/ICCV.2011.6126356

[3] Ernesto Brau and Hao Jiang. 2016. 3D Human Pose Estimation via Deep Learning from 2D Annotations. *International Conference on 3D Vision* (2016). https://doi.org/10.1109/3DV.2016.84

[4] Emmanuel J. Candès and Benjamin Recht. 2009. Exact matrix completion via convex optimization. *Foundations of Computational Mathematics* 9, 6 (2009), 717–772. https://doi.org/10.1007/s10208-009-9045-5 arXiv:0805.4471

[5] Zhe Cao, Tomas Simon, Shih-En Wei, and Yaser Sheikh. 2017. Realtime Multi-Person 2D Pose Estimation using Part Affinity Fields. (2017). https://doi.org/10.1109/CVPR.2017.143 arXiv:1611.08050

[6] Simone Ciotti, Edoardo Battaglia, Iason Oikonomidis, Alexandros Makris, Aggeliki Tsoli, Antonio Bicchi, Antonis A Argyros, and Matteo Bianchi. 2016. Synergy-driven Performance Enhancement of Vision-based 3D Hand Pose Reconstruction. In *International Conference on Wireless Mobile Communication and Healthcare (MobiHealth 2016), special session on advances in soft wearable technology for mobile-health*. 1–8. https://doi.org/10.1007/978-3-319-58877-3

[7] Petros Douvantzis, Iason Oikonomidis, Nikolaos Kyriazis, and Antonis A Argyros. 2013. Dimensionality Reduction for Efficient Single Frame Hand Pose Estimation. In *International Conference on Computer Vision Systems (ICVS 2013)*. Springer, St. Petersburg, Russia, 143–152.

[8] A. Elhayek, E. De Aguiar, A. Jain, J. Thompson, L. Pishchulin, M. Andriluka, C. Bregler, B. Schiele, and C. Theobalt. 2017. MARCOnI - ConvNet-Based MARker-less motion capture in outdoor and indoor scenes. *Pattern Analysis and Machine Intelligence, IEEE Transactions on* 39, 3 (2017), 501–514. https://doi.org/10.1109/TPAMI.2016.2557779

[9] Ali Erol, George Bebis, Mircea Nicolescu, Richard D. Boyle, and Xander Twombly. 2007. Vision-based hand pose estimation: A review. *Computer Vision and Image Understanding* 108, 1-2 (2007), 52–73. https://doi.org/10.1016/j.cviu.2006.10.012 arXiv:1412.0065

[10] Shachar Fleishman, Mark Kliger, Alon Lerner, and Gershom Kutliroff. 2015. ICPIK: Inverse Kinematics based articulated-ICP. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops*, Vol. 2015-Octob. 28–35. https://doi.org/10.1109/CVPRW.2015.7301345

[11] Michael Foukarakis, Ilia Adami, Danae Ioannidi, Asterios Leonidis, Damien Michel, Ammar Qammaz, Konstantinos Papoutsakis, Margherita Antona, and Antonis A Argyros. 2016. A Robot-based Application for Physical Exercise Training. In *International Conference on Information and Communication Technologies for Ageing Well and e-Health (ICT4AWE 2016)*. Scitepress, Rome, Italy, 45–52.

[12] Berthold K. P. Horn. 1987. Closed-form solution of absolute orientation using unit quaternions. *Journal of the Optical Society of America A* 4, 4 (1987), 629. https://doi.org/10.1364/JOSAA.4.000629

[13] Eldar Insafutdinov, Leonid Pishchulin, Bjoern Andres, Mykhaylo Andriluka, and Bernt Schiele. 2016. Deepercut: A deeper, stronger, and faster multi-person pose estimation model. In *European Conference on Computer Vision*. https://doi.org/10.1007/978-3-319-46466-4_3 arXiv:1605.03170

[14] Sam Johnson and Mark Everingham. 2011. Learning effective human pose estimation from inaccurate annotation. *Computer Vision and Pattern Recognition* (2011), 1465–1472. https://doi.org/10.1109/CVPR.2011.5995318

[15] Hanbyul Joo, Tomas Simon, and Yaser Sheikh. 2018. Total Capture: A 3D Deformation Model for Tracking Faces, Hands, and Bodies. *arXiv preprint arXiv:1801.01615* (2018). arXiv:1801.01615

[16] Nikolaos Kyriazis and Antonis A Argyros. 2013. Physically Plausible 3D Scene Tracking: The Single Actor Hypothesis. In *IEEE Computer Vision and Pattern Recognition (CVPR 2013)*. IEEE, Portland, Oregon, USA, 9–16. https://doi.org/10.1109/CVPR.2013.9

[17] Yann Lecun, Yoshua Bengio, and Geoffrey Hinton. 2015. Deep learning. *Nature* 521, 7553 (2015), 436–444. https://doi.org/10.1038/nature14539 arXiv:arXiv:1312.6184v5

[18] Ita Lifshitz, Ethan Fetaya, and Shimon Ullman. 2016. Human Pose Estimation using Deep Consensus Voting. In *European Conference on Computer Vision*. https://doi.org/10.1007/978-3-319-46475-6_16 arXiv:1603.08212

[19] Zhouchen Lin, Minming Chen, and Yi Ma. 2010. The Augmented Lagrange Multiplier Method for Exact Recovery of Corrupted Low-Rank Matrices. (2010). https://doi.org/10.1016/j.jsb.2012.10.010 arXiv:1009.5055

[20] Dushyant Mehta, Srinath Sridhar, Oleksandr Sotnychenko, Helge Rhodin, Mohammad Shafiei, Hans-Peter Seidel, Weipeng Xu, Dan Casas, and Christian Theobalt. 2017. VNect: Real-time 3D Human Pose Estimation with a Single RGB Camera. In *ACM Transactions on Graphics (SIGGRAPH 2017)*. https://doi.org/10.1145/3072959.3073596 arXiv:1705.01583

[21] Stan Melax, Leonid Keselman, and Sterling Orsten. 2013. Dynamics Based 3D Skeletal Hand Tracking. In *Proc. of Graphics Interface*. arXiv:1705.07640 http://arxiv.org/abs/1705.07640

[22] Damien Michel and Antonis A Argyros. 2016. Apparatuses, methods and systems for recovering a 3-dimensional skeletal model of the human body. (24 March 2016).

[23] Damien Michel, Ammar Qammaz, and Antonis A Argyros. 2017. Markerless 3D Human Pose Estimation and Tracking based on RGBD Cameras: an Experimental Evaluation. In *International Conference on Pervasive Technologies Related to Assistive Environments (PETRA 2017)*. ACM, Rhodes, Greece, 115–122.

[24] Thomas B. Moeslund and Erik Granum. 2001. A survey of computer vision-based human motion capture. *Computer Vision and Image Understanding* 81, 3 (2001), 231–268. https://doi.org/10.1006/cviu.2000.0897

[25] Francesc Moreno-Noguer. 2016. 3D Human Pose Estimation from a Single Image via Distance Matrix Regression. In *Computer Vision and Pattern Recognition*. https://doi.org/10.1109/CVPR.2017.170 arXiv:1611.09010

[26] Alejandro Newell, Kaiyu Yang, and Jia Deng. 2016. Stacked Hourglass Networks for Human Pose Estimation. In *European Conference on Computer Vision*. https://doi.org/10.1007/978-3-319-46484-8_29 arXiv:1603.06937

[27] Markus Oberweger and Vincent Lepetit. 2017. DeepPrior++: Improving Fast and Accurate 3D Hand Pose Estimation. In *ICCV workshop*, Vol. 840. https://doi.org/10.1109/ICCVW.2017.75 arXiv:1708.08325

[28] Markus Oberweger, Paul Wohlhart, and Vincent Lepetit. 2015. Hands Deep in Deep Learning for Hand Pose Estimation. In *Computer Vision Winter Workshop*. https://doi.org/10.1109/ICCV.2015.379 arXiv:1502.06807

[29] Ferda Ofli, Rizwan Chaudhry, Gregorij Kurillo, Rene Vidal, and Ruzena Bajcsy. 2013. Berkeley MHAD: A comprehensive Multimodal Human Action Database. *Proceedings of IEEE Workshop on Applications of Computer Vision* (2013), 53–60. https://doi.org/10.1109/WACV.2013.6474999

[30] OptiTrack. [n. d.]. OptiTrack - Motion Capture Systems. ([n. d.]). http://optitrack.com/

[31] Art B. Owen and Patrick O. Perry. 2009. Bi-cross-validation of the SVD and the nonnegative matrix factorization. *Annals of Applied Statistics* 3, 2 (2009), 564–594. https://doi.org/10.1214/08-AOAS227 arXiv:0908.2062

[32] Paschalis Panteleris and Antonis A Argyros. 2016. Monitoring and Interpreting Human Motion to Support Clinical Applications of a Smart Walker. In *Workshop on Human Motion Analysis for Healthcare Applications (HMAHA 2016)*. IET, London, UK.

[33] Georgios Pavlakos, Xiaowei Zhou, Konstantinos G. Derpanis, and Kostas Daniilidis. 2016. Coarse-to-Fine Volumetric Prediction for Single-Image 3D Human Pose. In *Computer Vision and Pattern Recognition*. https://doi.org/10.1109/CVPR.2017.139 arXiv:1611.07828

[34] Microsoft Corp. Redmond. [n. d.]. Kinect for Xbox 360. ([n. d.]).

[35] Konstantinos Roditakis, Alexandros Makris, and Antonis A Argyros. 2017. Generative 3D Hand Tracking with Spatially Constrained Pose Sampling. In *British Machine Vision Conference (BMVC 2017)*. BMVA, London, UK.

[36] Javier Romero, Dimitrios Tzionas, and Michael J. Black. 2017. Embodied hands: Modeling and Capturing Hands and Bodies Together. *ACM Transactions on Graphics (SIGGRAPH Asia 2017)* 36, 6 (2017), 1–17. https://doi.org/10.1145/3130800.3130883

[37] Jamie Shotton, Andrew Fitzgibbon, Mat Cook, Toby Sharp, Mark Finocchio, Richard Moore, Alex Kipman, and Aandrew Blake. 2011. Real-Time Human Pose Recognition in Parts from Single Depth Images. *Computer Vision and Pattern Recognition* (2011). http://research.microsoft.com/pubs/145347/BodyPartRecognition.pdf

[38] E Simo-Serra, C Torras, and F Moreno-Noguer. 2015. Lie algebra-based kinematic prior for 3D human pose tracking. *Machine Vision Applications (MVA), International Conference on* (2015), 0–3.

[39] Ayan Sinha, Chiho Choi, and Karthik Ramani. 2016. DeepHand: Robust Hand Pose Estimation by Completing a Matrix Imputed with Deep Features. In *Computer Vision and Pattern Recognition*. 4150–4158. https://doi.org/10.1109/CVPR.2016.450 arXiv:arXiv:1011.1669v3

[40] Xiao Sun, Yichen Wei, Shuang Liang, Xiaoou Tang, and Jian Sun. 2015. Cascaded hand pose regression. In *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, Vol. 07-12-June. 824–832. https://doi.org/10.1109/CVPR.2015.7298683

[41] D. Tang, J. Taylor, P. Kohli, C. Keskin, T. K. Kim, and J. Shotton. 2015. Opening the Black Box: Hierarchical Sampling Optimization for Estimating Human Hand Pose. In *ICCV*. 3325–3333. https://doi.org/10.1109/ICCV.2015.380

[42] Bugra Tekin, Isinsu Katircioglu, Mathieu Salzmann, Vincent Lepetit, and Pascal Fua. 2016. Structured Prediction of 3D Human Pose with Deep Neural Networks. In *British Machine Vision Conference*. arXiv:1605.05180 http://arxiv.org/abs/1605.05180

[43] Bugra Tekin, Artem Rozantsev, Vincent Lepetit, and Pascal Fua. 2016. Direct Prediction of 3D Body Poses from Motion Compensated Sequences. In *Computer Vision and Pattern Recognition*. https://doi.org/10.1109/CVPR.2016.113 arXiv:1511.06692

[44] Jonathan Tompson, Murphy Stein, Yann Lecun, and Ken Perlin. 2014. Real-Time Continuous Pose Recovery of Human Hands Using Convolutional Networks. *ACM Transactions on Graphics (SIGGRAPH 2014)* 33, 5 (2014), 1–10. https://doi.org/10.1145/2629500

[45] Dimitrios Tzionas, Luca Ballan, Abhilash Srikantha, Pablo Aponte, Marc Pollefeys, and Juergen Gall. 2016. Capturing Hands in Action Using Discriminative Salient Points and Physics Simulation. *International Journal of Computer Vision* 118, 2 (2016), 172–193. https://doi.org/10.1007/s11263-016-0895-4 arXiv:1506.02178

[46] Vicon. [n. d.]. Motion Capture Systems | Vicon. ([n. d.]). https://www.vicon.com/

[47] Robert Y. Wang and Jovan Popović. 2009. Real-time hand-tracking with a color glove. *ACM Transactions on Graphics* 28, 3 (jul 2009), 1. https://doi.org/10.1145/1531326.1531369

[48] Tao Yu, Kaiwen Guo, Feng Xu, Yuan Dong, Zhaoqi Su, Jianhui Zhao, Jianguo Li, Qionghai Dai, and Yebin Liu. 2017. BodyFusion : Real-time Capture of Human Motion and Surface Geometry Using a Single Depth Camera. In *International Conference on Computer Vision*. https://doi.org/10.1109/ICCV.2017.104

[49] Xingyi Zhou, Qixing Huang, Xiao Sun, Xiangyang Xue, and Yichen Wei. 2017. Towards 3D Human Pose Estimation in the Wild: a Weakly-supervised Approach. In *International Conference on Computer Vision*. https://doi.org/10.1109/ICCV.2017.51 arXiv:1704.02447