Freie Universität Berlin
Fachbereich Mathematik und Informatik
Takustraße 9, 14195 Berlin

MASTER THESIS

USER POSITION PREDICTION IN 6-DOF MIXED REALITY APPLICATIONS USING ARTIFICIAL RECURRENT NEURAL NETWORK

VORHERSAGE DER BENUTZERPOSITION IN 6-DOF-MIXED-REALITY-ANWENDUNGEN UNTER VERWENDUNG EINES KÜNSTLICHEN REKURRENTEN NEURONALEN NETZWERKS

Oleksandra Baga

Freie Universität Berlin
Matrikelnummer 5480722
Master Computer Science
E-Mail: oleksandra.baga@gmail.com

Prof. Dr. First Supervisor
Fachbereich Mathematik und Informati
Freie Universität Berlin

Prof. Dr. Second Supervisor
Fachbereich Mathematik und Informati
Freie Universität Berlin

Statutory Declaration

I herewith formally declare that I have written the submitted master thesis independently. I did not use any outside support except for the quoted literature and other sources mentioned in the paper.

I clearly marked and separately listed all of the literature and all of the other sources which I employed when producing this academic work, either literally or in content.

I am aware that the violation of this regulation will lead to failure of the thesis.

29.06.2022..... Oleksandra Baga

Acknowledgments

This thesis was created in cooperation with the Fraunhofer Heinrich Hertz Institute.

First and foremost I would like to thank Prof FU Berlin, who supervised my thesis. Thank you very much for the helpful suggestions and constructive criticism.

A special thanks goes to the researcher Fraunhofer Heinrich Hertz Institute, Serhan Gül, who suggested an exciting topic for a research, which I was allowed to choose for my master thesis. I would like to express my sincere thanks for the commitment and the consultation during the preparation of this thesis.

Contents

List of Figures

Listings

List of Abbreviations

ANN	Artificial Neural Networks
AR	Augmented Reality
CNN	Convolutional Neural Network
CPU	Central processing unit
DoF	Degree of freedome
DL	Deep Learning
FFN	Feed-forward Neural Network
GRU	Gated Recurrent Unit
HMD	Head-Mounted-Display
IEEE	Institute of Electrical and Electronics Engineers
KF	Kalman Filter
LAT	Look ahead time
LSTM	Long-Short-Term Memory
M2P	Motion-to-Photon
MAE	Mean Absolut Error
MEC	Mobile Edge Computing
ML	Machine Learning
MR	Mixed Reality
NLP	Natural Language Processing
ReLu	Rectified Linear Unit
RNN	Recurrent Neural Network
RTT)	Round-trip time
SDG	Stochastic Gradient Descent
VR	Virtual Reality
3-DoF	Three degree of freedom
6-DoF	Six degree of freedom

1 Introduction

This thesis is focusing on designing and evaluation of the approach for the prediction of human head position in a 6-dimensional degree of freedom (6-DoF) of Extended Reality (XR) applications for a given look-ahead time (LAT) in order to reduce the Motion-to-Photon (M2P) latency of the network and computational delays. At the beginning of the work the existing 3-DoF as well as 6-DoF methods were analyzed, and their similarities differences were taken into account when a proposed Recurrent Neural Network-based predictor was developed. The investigation of different neural network architectures and the improvement of head motion prediction is the main goal of this thesis. Proposed approach was evaluated and at the end of the work the obtained results were discussed and the suggestions for future work were done.

1.1 Problem statement

The correct and fast head movement prediction is a key to provide a smooth and comfortable user experience in VR environment during head-mounted display (HDM) usage. The recent improvements in computer graphics, connectivity and the computational power of mobile devices simplified the progress in Virtual Reality (VR) technology. The way users can interact with their devices changed dramatically. With new technologies of VR environment user becomes the main driving force in deciding which portion of media content is being displayed to them at any time of interaction with VR Applications [19]. Until recently the high-quality experiences with modern Augmented Reality (AR) and VR systems were not widely presented in home usage and were mainly used in research labs or commercial setups. The hardware for displaying the VR environment was once extremely expensive but recent years became more broadly accessible and the 6-DoF VR headset designed for the end-user were released[1]. It is possible now to experience virtual reality scenes and watch new type of volumetric media at home and the market interest for development VR and AR applications expected to be huge next years.
In fact, the existing on this moment virtual environments can be divided into two main groups depending on position of the user and their ability to move inside the VR environment. The user motion and prediction within a 3-DoF environment has been intensely researched for years. Extending such approaches to a 6-DoF

[1] https://medium.com/@DAQRI/motion-to-photon-latency-in-mobile-ar-and-vr-99f82c480926

environment is not straightforward, due to the change of the user's viewing point from inward to outward and additional three degrees of freedom [20].
Although all mentioned above improvements, rendering of volumetric content remains very demanding task for existing devices. Thus the improvement of a performance of existing methods, design and implementation of new approaches specially for the 6-DoF environment could be a promising research topic.

1.2 Motivation for the research

Research efforts to reduce the computational load are being already wide attempted. However, these approaches designed for the client side. Recently presented technique of the rendering on a cloud server makes possible to decrease the computational load on the client device by offloading of the task to a server infrastructure and than by sending the rendered 2D content instead of volumetric data [13]. The calculated 2D view must correspond the current position and orientation of a user. However, cloud-based streaming approach adds network latency and processing delays due uploading to a server the user position, rendering a new 2D picture from the 3D data and sending it back to a device. Thus, a rendered 2D image can appear even later on a display than with usage of local rendering system.
The promising research topic is reducing the Motion-to-Photon (M2P) latency by predicting the future user position and orientation for a look-ahead time (LAT) and sending the corresponding rendered view to a client. The LAT in this approach must be equal or larger to the M2P latency of the network including round-trip time (RTT) and time need for calculation and rendering of a future picture at remote server.

1.3 Structure of the thesis

The organization of this thesis is as follows. The thesis starts from introduction and problem statement, followed by theoretical background related to the research topic. Literature review chapter introduces different approaches and technologies of motion prediction algorithms. The chapters 4 and 5 show the implementation of presented model and evaluation of the results that were obtained during experiments. Last, the discussion regarding method limitations and suggestions for the future work are done.

Chapter 1 - Introduction.
The current chapter shortly introduces a state of development on scientific field achieved at a time of master thesis creation in the context on XR applications. The necessity of timely action to improve the situation with increasing computational

and network latency is shown in problem statement section 1.1. Due to the breadth of the research topic, the section 1.2 focuses the research topic and clearly motivates the implementation with neural network model.

Chapter 2 - Background.
The next chapter includes a review of the area being researched. It starts with a short introduction of the concept of MR applications and presents a 6-DoF environment. The presence and influence of a computational and network latency is covered, followed by discussion of possible solutions for its reduction. In section 2.5 the head pose estimation algorithms and the challenges faced in predicting of the viewer's position are discussed.

Chapter 3 - Related work.
The chapter provides an overview of previous research in the field of prediction of user's head position and orientation. The related works divided into section corresponding the computational approach of the prediction method. The chapter places a master thesis's topic in the context of the existing literature and the last section focuses in the methods using neural networks.

Chapter 4 - Data and Model.
Fourth chapter presents the design of the proposed method. The dataset including data collection from head mounted display (HDM) and data understanding and preprocessing are described in section 4.1, followed by a design of the algorithm including network architecture, functions of an input layers and the training methods.

Chapter 5 - Data and Model.
Fifth chapter gives an overview of the implementation of the described in previous chapter method. It describes the conducted experiments with a data obtained from HMD. The evaluation metrics and results are presented in the section 5.3.

Chapter 6 - Analysis.
The last chapter presents a discussion about the limitation of proposed method and provides a conclusion about the received results including suggestions for potential types of future research.

2 Background

This chapter introduces theoretical background of the presented research problem. First, the concept of mixed reality (MR) and the relation of this huge topic to the research field is presented, followed by an introduction of six degree of freedom (6-DoF) environment and the difference to the three degree of freedom (3-DoF). The term motion-to-photon latency (M2P) is covered, followed by a short discussion about an influence of M2P latency on the decreasing of user experience. The new developed cloud-based rendering and streaming approach is shortly discussed in this chapter. The last section of this chapter highlights challenges with the prediction of viewer's head pose that arises in modern XR applications in connection especially with the added network latency due the using of remote cloud server for computational offload.

2.1 Mixed reality

2.2 Six degrees of freedom

2.3 Motion-to-photon latency

2.4 Cloud-based volumetric video streaming

2.5 Challenges of head motion prediction

All modern HMD has a position tracker, a device or a system of devices, that is responsible for reporting the position and orientation of HMD to the computational unit that generates the virtual environment images displayed in the HMD. These images represent the view that a wearer of HMD would have seen if user was present in VR at the position and orientation reported by position tracker [6].
While the task of position tracking is performed by HMD hardware, the task of position prediction of the movement of human body in the virtual reality remains challenging, and it is still complicate to achieve high-precision estimation. Recurrent

neural networks have recently shown promising results in many machine learning tasks, especially when input and/or output are of variable length and are coming as time series with a sequential order. Unfortunately, the known problem of RNN that was observed many years ago by, e.g., *Bengio et al., 1994* that it is difficult to train RNNs to capture long-term dependencies because the gradients tend to either vanish (most of the time) or explode (rarely, but with severe effects) [5]. New approaches are needed to be implemented to reduce the negative impacts of this issue. Since traditional recurrent unit overwrites its content at each time-step, a LSTM unit is able to decide whether to keep the existing memory via the introduced gates. The Long Short-Term Memory (LSTM) has a number of minor modifications [9] since it was initially proposed in work [14]. Another approach called a gated recurrent unit (GRU) can adaptively capture dependencies of different time scales without having a separate memory cells [9]. These two approaches can help to find the long-term dependencies in the data obtained from HMD that are otherwise are hidden by the effect of short-term dependencies from the standard RNN models.

More challenges and problems.

3 Related work

This chapter presents the overview of previous research in the field of the prediction of user position. It includes both approaches for 3-DoF and 6-DoF environment, focusing on time series methods, Kalman Filter and overview of different Recurrent Neural Network architectures such as LSTM and GRU.

3.1 Time series methods

3.2 Kalman Filter

KF-based extrapolation are deemed to be robust against fast fluctuations, but suffer from susceptibility to noise sensory data [3].

3.3 Recurrent Neuronal Network

In this section the network architectures of most relevant works in the field head motion prediction with RNNs are explained and discussed whether LSTM or GRU architecture is better for the actual problem presented in this thesis. As was mentioned in section 3.1, typical HMD computes user positions in 6-DoF by using its tracker module and data comes as time series with a sequential order. The structure of an input is crucial and needed to be followed in order to predict the next future step for a look-ahead time correctly. A sequence of inputs can be processed with Artificial Neural Network (ANN) called Recurrent Neural Network (RNN). Moreover, RNN can processes input with remembering its state while processing the next sequence of inputs. In the last decade, RNN algorithms have been adopted for motion prediction of 3D sequences. For example, the work of *Crivellari et al., 2020* targets traces of short-term tourists in a foreign country and tries to predict the motion of people in the environment they never seen before. LSTM-based model is used thus for analyzing the tourists' mobility patterns [10]. The work of *Yang et al., 2020* proposes using of flashbacks on hidden states in RNNs to search past hidden states with high predictive power. Authors mentioned that their approach gives an ability to determine a specific Point of Interests (POIs) from user's mobility

trace stored as a sequence of check-ins in Based Social Networks (LBSNs). Exploring the dataset researchers found sparsity and incompleteness of input sequences and thus the sequential pattern is difficult to be captured by RNNs. They considered temporal distances between similar POIs by flashing back to the historical hidden states sharing a similar temporal context as the current one [23].

The authors *Aykut et al., 2018* claims their research to be first work that applies deep learning for head motion prediction. The current three rotations in three dimensions the so-called Euler angles as well as past values thereof within a certain time window (W) used as input for the network [3]. The best results delivered when 20 last values for each orientation direction (W = 250 ms) were used [3]. Instead of the absolute orientation values *Aykut et al., 2018* suggest for a goal of generalization for other datasets to subdivide the inputs into their respective orientation groups and compute their normalized differences [3]. The authors experimentally confirmed that Feed-forward Neural Network (FFN) indeed had difficulties to learn for different delays even after an architecture was extended with the present delay and injected into all NN nodes. The using of LSTM-based architectures *Aykut et al., 2018* reasoned with feedback loop and ability to establish a way of memory and share weights over time [3]. Researchers used Adam optimization algorithm, the maximum number of epochs was set to 1000, early stopping technique (patience = 2, min. delta = 0) was used to avoid overfitting. Additionally, the learning rate was decreased by 70% from initial value of 0.001 every 30 epochs. The batch size B was set to 2^{11}. Rectified Linear Unit (ReLU) as activation function for the FFN layers used with LSTM [3]. Three different architecture were tried. In all variants an input for NN was subdivided into dimensions and normalized within time window W set to 250 ms with $\triangle$t = 25 ms. Thus the length of input vector is equal to 10. Final result obtained from each NN variant is a vector of length of 10 for each pan, tilt and roll dimension separately containing future prediction with LAT 1s and step of 0.1s.

Conducted by researchers experiments showed that the LSTM-based architecture leads to a significant improvement of the MAE and RMSE metrics. The best performance is achieved by the interleaved architecture of LSTM and dense FFN blocks [3]. The LSTM-based methods were compared also to widely used approaches like the Linear Regression and a Kalman Filter based optimal state estimate. Thus *Aykut et al., 2018* demonstrated a substantial improvement of the deep predictor for latencies in the range of 0.1–0.9 s [3].

Next year *Aykut et al., 2019* experimented in their work [4] with GRU model that belongs to the group of recurrent neural networks (RNN). Authors mentioned that they already achieved best performance for headmotion prediction with LSTM as described before. However they tried an usage of dense GRUs combined with convolutional components regarding that fact that GRU is computationally more efficient, as it has fewer parameters and states than LSTM units [4]. The GRU-CNN-based network also receives pan, tilt, and roll present and past values within the same W = 250ms for last 20 values. Network trained on the normalized differences

instead of using of the absolute values as it was done in [3]. A convolution layer performs as low-pass filter and reduces the noise. The core of the network consists of an interleaved structure of six dense GRUs each with 30 nodes and subsequent convolution units (kernel size 30x30), which compute the most distinct features. Max pooling layer down-samples the output of the last convolution layer and keeps the most significant features. The last step is passing the results through dense FFN and inverse remapping to the absolute orientation values. The model also predicts whole sequence of orientation values with LAT 1s and step of 0.1s as in [4]. The number of epoch in training remains the same as in [3] but early-stopping technique has higher patience parameter and equal 10. Compared to [3] the batch size was reduced to 2^9. The rest of model parameters remains the same as in previous work of *Aykut et al., 2018*. Authors said that proposed GRU-CNN-based network is able to improve the MAE and RMSE compared to all Kalman Filter and described above LSTM-CNN methods, especially for larger delays [4].

Researchers *Karim et al., 2018* developed long short term memory fully convolutional network (LSTM-FCN). In the proposed models, the fully convolutional block is augmented by an LSTM block followed by dropout. The fully convolutional block consists of three stacked temporal convolutional blocks with filter sizes of 128, 256, and 128 respectively [15]. Each convolutional block is identical to the convolution block in the CNN architecture proposed by *Wang et al., 2018* in their work [22]. This means the basic block is a convolutional layer followed by a batch normalization layer and a ReLU activation layer. The convolution operation is fulfilled by three 1-D kernels with the sizes 8, 5, 3 without striding [22]. *Wang et al., 2018* excluded in their FCN any pooling operation to prevent overfitting. Batch normalization helps to reduce generalization error. Global average pooling layer reduces the number of weights. Softmax layer produces the final labels [22]. The LSTM block, comprising of either a general LSTM layer or an Attention LSTM layer, is followed by a dropout. The output of the global pooling layer from FCN output and the LSTM block is concatenated and passed onto a softmax classification layer [15].

The important detail is that FCN block and LSTM block perceive the same time series input in two different views. If there is a time series of length N, the fully convolutional block will receive the data in N time steps [15]. *Karim et al., 2018* sends additionally the time series input into a dimension shuffle layer. Thus LSTM block receives transformed input having N variables with a single time step. Authors say that without the dimension shuffle, the performance of the LSTM block is significantly reduced due to the rapid overfitting of small short-sequence [15].

In work of *Chang et al., 2020* LSTM-based model is proposed to predict displacement of positions in each timestamp given measured acceleration and Euler angles from sensor signals [7]. In addition to standard LSTM networks, they also use bidirectional LSTM (Bi-LSTM) networks, which is stacked two LSTM networks in forward and backward directions. Standard LSTM networks can only consider the past information and Bi-LSTM networks can capture both past and future informa-

tion by two opposite temporal order in hidden layers [7]. Authors predicted first displacement $x_{t_2} - x_{t_1}$ over the time interval $[t1, t2]$ and used it to obtain the position. Experimentally, authors found that the basic LSTM performs the best comparing to Bi-LSTM and Temporal Convolutional Network. Thus LSTM network with 3 layers used in this paper as the main structure.

The paper [16] also aims for action recognition using sensor signals from HMD. This approach presents another combination of of gated recurrent unit and additional model. Convolutional Auto-Encoder (CAE) model in a clustering fashion learns discriminative feature representation and helps to conquer difficulty of recognizing similar actions. As in the work [7], a deep sequential model learns the temporal representations and predicts the displacement of positions with acceleration and Euler angle to reduce error accumulation. Similar to [7], researches also used additionally a bidirectional LSTM (Bi-LSTM) network. The usage of CAE is proposed as a feature extraction model for feature representation of action signals and is composed of an encoder and decoder [16]. The encoder contains two 2D convolutional layers with a kernel of size 3×1 with the stride of 1. The decoder contains two 2D transposed convolutional layers with the same kernel size and stride. The bottleneck is three feed-forward fully-connected layers with neurons of 1024, 128, and 1024, respectively. The tanh function is used as the activation function between the convolution layer and the fully connected layer [16]. Input contains body acceleration, gravity acceleration, gyroscope, and quaternion with dimension equal to 26. The action signal is divided into segments of 25 timestamps by stride-1 sliding windows In this approach an action is performed and recorded separately and ground-truth labels was assigned. Then, the signals of each action with ground-truth labeling are used as input to CAE model for feature extraction by sliding windows [16]. Authors mentioned a representation of a signal data by the latent vector in the low-dimensional space after using the CAE model. The latent vector then will be clustered by K-Means Clustering so that centroids are referred to as motion bases in a model. For action tracking with LSTM, the displacement in each timestamp was predicted first and then added to its previous location, instead of predicting each position directly [16]. Similar as in work [7] the 3-layered LSTM model performed better compared to Bi-LSTM. Authors said that the possible reason could be the short-term correlation of human actions in their dataset and that Bi-LSTM with its complicated model structure is rather suitable for long-term actions [16].

The paper of *Chung et al., 2014* should be noted separately in the end of related works review because it provides an interesting compassion and evaluation of the performance of recurrent units LSTM and GRU on sequence modeling. Authors mentioned the ability of LSTM to keep the existing memory via the introduced gates and thus to detect an important feature from an input sequence at early stage, to easily carry this information (the existence of the feature) over a long distance, hence, capturing potential long-distance dependencies [9]. The GRU takes linear sum between the existing state and the newly computed state similar to the LSTM

but does not have any mechanism to control the degree to which its state is exposed, but exposes the whole state each time [9]. *Chung et al., 2014* emphasize the fact that any important feature, decided by either the forget gate of the LSTM unit or the update gate of the GRU, will not be overwritten but be maintained as it is [9]. LSTM unit controls the amount of the new memory content and does not have any separate control of the amount of information flowing from the previous time step. The GRU differs and controls the information flow from the previous activation when computing the new and does not independently control the amount of the candidate activation being added via update gate [9]. The experiments provided in this work clearly indicate the advantages of the gating units over the more traditional recurrent units. *Chung et al., 2014* mentioned that with dataset they used GRU unit outperformed LSTM unit. But they suggest that the choice of the type of gated recurrent unit may depend heavily on the dataset and corresponding task [9]. Thus in section 4.1.2 of chapter "Data and Model" the data exploration of the obtained from HMD Microsoft HoloLend is done in order to understand the dataset before the beginning with a development of model architecture.

4 Data and Model

4.1 6-DoF Dataset

4.1.1 Data collection from HMD

The low-cost IMU is sometimes unstable in frame rate during collecting data. In the experiments, we set the frame rate of IMU with 50 Hz which means that data will be collected per 0.02 second. Unfortunately, sometimes IMU could occur delay, and the time gap of two samples may be reduced to 0.01 second or increased to 0.05 second. To deal with above situation,

4.1.2 Data Exploration

!! Data analysis AVG linear velocity position, plots

4.1.3 Data preprocessing

4.2 Neural Network

4.2.1 Network architecture

4.2.2 Network input

4.3 Training methods

Implementation and experiments

5

Here some code for my super neural network. The artificial neural networks discussed in this text are only remotely related to their biological counterparts. In this section we will briefly describe those characteristics of brain function that have inspired the development of artificial neural networks.

```
class StudentFactory(DjangoModelFactory):
        class Meta:
                model = Student
        student_card = factory.SubFactory(StudentCardFactory)
        first_name = factory.Faker('first_name')
        second_name = factory.Faker('last_name')
```

Listing 5.1: StudentFactory

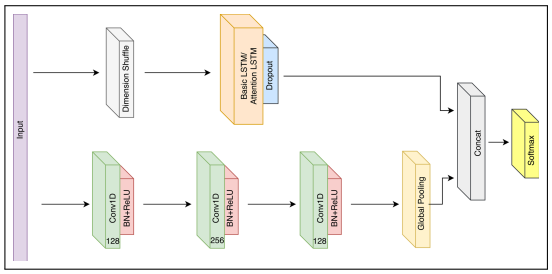

Figure 1: LSTM Fully Convolutional Networks for Time Series Classification

You might already know that you want to apply an established theory or set of theories to a specific context (for example, reading a literary text through the lens of critical race theory, or using social impact theory in a market research project).

5.1 Implementation

5.2 Experiments

5.3 Evaluation metrics

5.4 Results

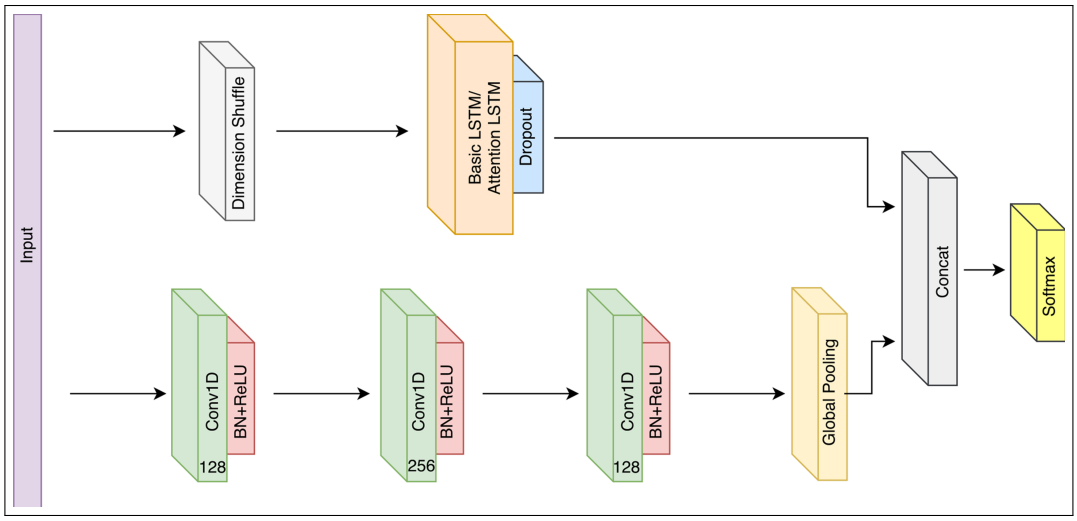

Figure 2: LSTM FCN WRAP

6 Analysis

The introduction text to analysis chapter

6.1 Limitations

6.2 Conclusion

6.3 Suggestions for future work

Glossary

AJAX

AJAX (asynchrones Javascript und XML) ist der allgemeine Name für Technologien, mit denen asynchrone Anforderungen (ohne erneutes Laden von Seiten) an den Server gestellt und Daten ausgetauscht werden können. Da die Client- und Serverteile der Webanwendung in verschiedenen Programmiersprachen geschrieben sind, müssen zum Austausch von Informationen die Datenstrukturen (z. B. Listen und Wörterbücher), in denen sie gespeichert sind, in das JSON-Format konvertiert werden.

Bibliography

[1] A. Deniz Aladagli, Erhan Ekmekcioglu, Dmitri Jarnikov, and Ahmet Kondoz. *Predicting head trajectories in 360° virtual reality videos*. https://ieeexplore.ieee.org/document/8251913. date access on 29.03.22. 2017. DOI: 10.1109/IC3D.2017.8251913.

[2] R.S. Allison, L.R. Harris, M. Jenkin, U. Jasiobedzka, and J.E. Zacher. *Tolerance of temporal delay in virtual environments*. https://www.researchgate.net/publication/2945506. date access on 17.03.22. 2001. DOI: 10.1109/VR.2001.913793.

[3] Tamay Aykut, Christoph Burgmair, Mojtaba Leox Karimi, and Eckehard Steinbach. *Delay Compensation for a Telepresence System With 3D 360 Degree Vision Based on Deep Head Motion Prediction and Dynamic FoV Adaptation*. https://arxiv.org/abs/2007.14084. date access on 23.02.22. 2018. DOI: 10.1109/WACV.2018.00222.

[4] Tamay Aykut, Eckehard Steinbach, and Jingyi Xu. *Time Series Classification from Scratch with DeepNeural Networks: A Strong Baseline*. https://www.researchgate.net/publication/318332658. date access on 25.03.22. 2017. DOI: 10.1109/IJCNN.2017.7966039.

[5] Y. Bengio, P. Simard, and P. Frasconi. *Learning long-term dependencies with gradient descent is difficult*. http://www.cs.unc.edu/techreports/93-010/93-010.pdf. date access on 31.03.22. 1994. DOI: 10.1109/72.279181.

[6] Devesh K Bhatnagar. *Position trackers for Head Mounted Display systems: A survey*. http://www.cs.unc.edu/techreports/93-010/93-010.pdf. date access on 31.03.22. 1993.

[7] Yun-Kai Chang, Mai-Keh Chen, Yun-Lun Li, Hao-Ting Li, and Chen-Kuo Chiang. *6DoF Tracking in Virtual Reality by Deep RNN Model*. https://ieeexplore.ieee.org/document/9394069. date access on 02.04.22. 2020. DOI: 10.1109/IS3C50286.2020.00057.

[8] Howie Choset et al. *Principles of Robot Motion: Theory, Algorithms, and Implementations*. the MIT Press, 2006, p. 630. ISBN: 978-0262033275.

[9] Junyoung Chung, Caglar Gulcehre, KyungHyun Cho, and Yoshua Bengio. *Empirical Evaluation of Gated Recurrent Neural Networks on Sequence Modeling*. https://arxiv.org/abs/1412.3555. date access on 30.03.22. 2014. DOI: 10.48550/arXiv.1412.3555.

[10] Alessandro Crivellari and Euro Beinat. *LSTM-Based Deep Learning Model for Predicting Individual Mobility Traces of Short-Term Foreign Tourists*. https://www.researchgate.net/publication/338377314. date access on 08.04.22. 2020. DOI: 10.3390/su12010349.

[11] Richard O. Duda, Peter E. Hart, and David G. Stork. *Principles of Robot Motion: Theory, Algorithms, and Implementations*. Wiley-Interscience; 2. edition, 2000, p. 688. ISBN: 978-0471056690.

[12] Serhan Guel, Sebastian Bosse, Dimitri Podborski, Thomas Schierl, and Cornelius Hellge. *Kalman Filter-based Head Motion Prediction for Cloud-based Mixed Reality*. https://arxiv.org/abs/2007.14084. date access on 19.02.22. 2020. DOI: 0.1145/3394171.3413699.

[13] Serhan Gül, Dimitri Podborski, Thomas Buchholz, Thomas Schierl, and Cornelius Hellge. *Low-latency Cloud-based Volumetric Video Streaming Using Head Motion Prediction*. https://arxiv.org/abs/2001.06466. date access on 19.02.22. 2020. DOI: 0.1145/3394171.3413699.

[14] Sepp Hochreiter and Jürgen Schmidhuber. *Long Short-term Memory*. https://www.researchgate.net/publication/13853244. date access on 31.03.22. Dec. 1997. DOI: 10.1162/neco.1997.9.8.1735.

[15] Fazle Karim, Somshubra Majumdar, Houshang Darabi, and Shun Chen. *LSTM Fully Convolutional Networks for Time Series Classification*. https://arxiv.org/abs/1709.05206. date access on 14.04.22. 2017. DOI: 10.48550/arXiv.1709.05206.

[16] Hao-Ting Li, Yung-Pin Liu, Yun-Kai Chang, and Chen-Kuo Chiang. *Action recognition and tracking via deep representation extraction and motion bases learning*. https://www.researchgate.net/publication/358012181. date access on 01.04.22. 2022. DOI: 10.1007/s11042-021-11888-8.

[17] Anh Nguyen, Zhisheng Yan, and Klara Nahrstedt. *Your Attention is Unique: Detecting 360-Degree Video Saliency in Head-Mounted Display for Head Movement Prediction*. https://www.researchgate.net/publication/328370817. date access on 15.03.22. 2018. DOI: 10.1145/3240508.3240669.

[18] Feng Qian, Lusheng Ji, Bo Han, and Vijay Gopalakrishnan. *Optimizing 360 video delivery over cellular networks*. https://dl.acm.org/doi/10.1145/2980055.2980056. date access on 12.03.22. 2016.

[19] Silvia Rossi, Irene Viola, Laura Toni, and Pablo Cesar. *A New Challenge: Behavioural Analysis Of 6-DOF User When Consuming Immersive Media*. https://ieeexplore.ieee.org/document/9506525. date access on 03.04.22. 2021. DOI: 10.1109/ICIP42928.2021.9506525.

[20] Silvia Rossi, Irene Viola, Laura Toni, and Pablo Cesar. *From 3-DoF to 6-DoF: New Metrics to Analyse Users Behaviour in Immersive Applications*. https://www.researchgate.net/publication/357172010. 1-7. 2021. date access on 13.04.22.

[21] Sebastian Thrun, Wolfram Burgard, and Dieter Fox. *Probabilistic Robotics*. the MIT Press, 2006, p. 672. ISBN: 978-0262201629.

[22] Zhiguang Wang, Weizhong Yan, and Tim Oates. *Time Series Classification from Scratch with DeepNeural Networks: A Strong Baseline*. https://www.researchgate.net/publication/318332658. date access on 25.03.22. 2017. DOI: 10.1109/IJCNN.2017.7966039.

[23] Dingqi Yang, Benjamin Fankhauser, Paolo Rosso, and Philippe Cudre-Mauroux. *Location Prediction over Sparse User Mobility Traces Using RNNs: Flashback in Hidden States*. https://www.researchgate.net/publication/338377314. date access on 07.04.22. 2020. DOI: 10.24963/ijcai.2020/302.

[24] Emin Zerman, Radhika Kulkarni, and Aljosa Smolic. *User Behaviour Analysis of Volumetric Video in Augmented Reality*. https://ieeexplore.ieee.org/document/9465456. date access on 13.04.22. 2021. DOI: 10.1109/QoMEX51781.2021.9465456.