2024 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)
October 14-18, 2024. Abu Dhabi, UAE

Force and Velocity Prediction in Human-Robot Collaborative Transportation Tasks through Video Retentive Networks

J. E. Domínguez-Vidal and Alberto Sanfeliu

***Abstract*— In this article, we propose a generalization of a Deep Learning State-of-the-Art architecture such as Retentive Networks so that it can accept video sequences as input. With this generalization, we design a force/velocity predictor applied to the medium-distance Human-Robot collaborative object transportation task. We achieve better results than with our previous predictor by reaching success rates in testset of up to 93.7% in predicting the force to be exerted by the human and up to 96.5% in the velocity of the human-robot pair during the next $1\ s$, and up to 91.0% and 95.0% respectively in real experiments. This new architecture also manages to improve inference times by up to 32.8% with different graphics cards. Finally, an ablation test allows us to detect that one of the input variables used so far, such as the position of the task goal, could be discarded allowing this goal to be chosen dynamically by the human instead of being pre-set.**

***Index Terms*— Physical Human-Robot Interaction, Object Transportation, Force Prediction, Human-in-the-Loop**

I. Introduction

Robotics has always allowed us to put into practice and test in the real world the advances made in fields as diverse as automatic control, physics, psychology or artificial intelligence in general and Deep Learning in particular. In this way, we have improved the capabilities of robots to work autonomously with increasingly precision [1], [2] and, more recently, to collaborate with us humans in multiple tasks such as handover [3], collaborative search [4], [5], collaborative assembly [6], [7] among others.

In this work, we focus on the task of human-robot collaborative transportation (see Fig. 1). More specifically, it is the improvement of our previous work [8], [9]. While in [8] we proved that it was possible to take advantage of the fact that this is a task where information exchange occurs mainly through forces to develop a first predictor of the next force to be exerted by the human, in [9] we improved its architecture using Transformers [10] and extended its capabilities to also predict the velocity of the human-robot pair.

In this article we develop an improved version of our force and velocity predictor based on a more recent architecture such as Retentive Networks (RetNet) [11]. To this end, we first generalize these RetNets to be able to process video sequences and not only one-dimensional data sequences. To the best of our knowledge, this generalization has not been proposed in any other work, being this our first contribution, which can be useful for other tasks and even other fields. By means of this generalization, we managed to improve the performance of our predictor, both in accuracy when predicting force and velocity and obtaining inference times up to 32.8% lower with respect to the previous best version, being this our second and main contribution. Finally, in different tests performed, this new design of our predictor seems to be less sensitive to knowing *a priori* the location of the goal to which the human-robot pair should take the object, being this one of the main limiting factors of the previous versions, opening the door to this goal being chosen on the fly by the human regarding force prediction.

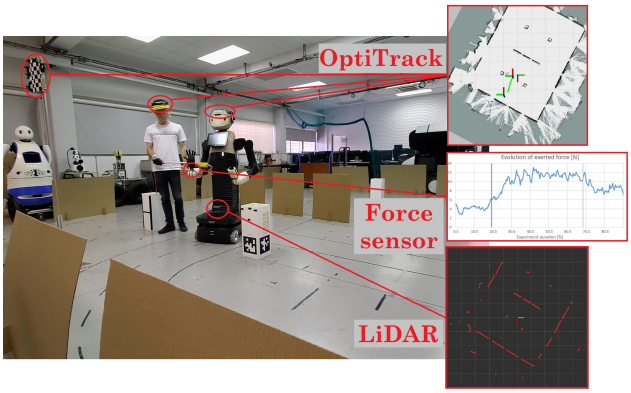

Fig. 1. **Collaborative transportation set-up.** Human and robot must collaborate to transport an aluminium bar until the goal (marked with a chequered flag) through a complex scenario with multiple routes. OptiTrack on the ceiling to detect and track the human, the robot and the goal for posterior analysis. Force sensor on robot's wrist to detect human's exerted force over the transported object. LiDAR to detect the environment and make an occupancy map. Details about how to interprest these data in Sec. III-A.

In the remainder of the article, Section II presents the related work. Section III presents the architecture of the force and velocity predictor presented in this article. Section IV shows the results obtained regarding the performance of the predictor both in dataset and real experiments. Finally, Section V presents the conclusions and future work.

II. Related Work

When we talk about joint manipulation of objects between a human and a robot, the first solutions that can be found in the literature are based on the use of admittance [12], [13] and impedance [14], [15] controllers, all of them aiming to make the robot adapt to the human's actions as fast as possible. While it is also possible to find works that attempt

Work supported under the European project CANOPIES (H2020-ICT-2020-2-101016906) and by JST Moonshot R & D Grant Number: JPMJMS2011-85. The first author acknowledges Spanish FPU grant with ref. FPU19/06582.

The authors are with the Institut de Robòtica i Informàtica Industrial (CSIC-UPC). Llorens Artigas 4-6, 08028 Barcelona, Spain and with Universitat Politècnica de Catalunya - BarcelonaTech (UPC). Jordi Girona, 31, 08034, Barcelona, Spain. {jdominguez, sanfeliu}@iri.upc.edu. The first one is the corresponding author.

to include as input to these controllers some kind of prediction of the trajectory of the human [16] or the object [17], it was not until the proliferation of Deep Learning models that these predictions improved sufficiently. Thus, [18] uses Reinforcement Learning (RL) to model the uncertainties of the human as an imput of a Model Predictive Controller (MPC) and [19], [20] rely on Learning from Demonstration (LfD) to learn the task to be executed or predict the human's desired speed profile.

Even though some of the previous work [20] has as input for its architecture the force exerted by the human, no work beyond our previous work [8], [9] attempts to obtain a prediction of the force that will be exerted by the human in the near future. Instead, they usually choose to try to predict the trajectory of the human or the transported object because it is more stable and, therefore, easier to predict. We, on the other hand, advocate predicting the force to be exerted because it allows us to detect changes in human's intention more quickly. Moreover, this prediction can be further processed to obtain a trajectory estimation [8].

While in [9] we processed both visual and sequential information based on well-known architectures such as Transformers [10] and its version oriented to process image sequences such as the Video Vision Transformer (ViViT) [21], in this work we will be inspired by Retentive Networks (RetNet) [11]. These RetNets, like Linear Transformers [22] or RWKV [23], seek to improve the performance of Transformers when performing inference, although, in the case of Linear Transformers, at the cost of worsening their overall performance. RetNets, on the other hand, seek to maintain the performance of a Transformer or even improve it by optimizing resource consumption. Although this type of network is used to process one-dimensional data sequences and [24] generalizes it to process images, our article is the first to perform a more extensive generalization that allows them to be used to process videos or sequences of images.

III. RetNet-based Force/Velocity Predictor for Collaborative Object Transportation

In order to compare this new predictor with our previous work [8], [9] and expand the dataset used, we use the same collaborative task (human-robot transport of objects) in the same scenario in which multiple obstacles are placed so that the human has at all times multiple routes available to get the object to its predefined destination and can even change routes on the fly (see Fig. 1).

A. Problem formulation

The goal is to obtain a prediction of the next T measurements of the force that the human will exert on one end of the transported object, $Y^{force}_{N+1:N+T} \in \mathbb{R}^{2,T}$, and of the following T values that the velocity of the human-robot pair will take, $Y^{vel}_{N+1:N+T} \in \mathbb{R}^{2,T}$. As discussed, the former prediction is useful for detecting rapid changes in the human's intention, while the latter allows us to obtain a better estimate of the trajectory that the pair will follow by taking into account both the human's and the robot's contribution to the task.

In order to obtain both predictions, we use five information inputs. First, we use the LiDAR+LaserScan of the robot to obtain an occupancy map of the environment. This map will be an image of 100x100 pixels in which each pixel will indicate whether the equivalent area of 10x10 cm in the real scenario is occupied or not. The second information source is the result of giving semantic meaning to this occupancy map. By clustering the occupied cells, we detect the O obstacles visible at each moment by the robot and assign to each of them a repulsive force inversely proportional to the distance between the obstacle and the pair, $\boldsymbol{f_{C,obs_i}} \in \mathbb{R}^2$. In parallel, the robot uses a global planner to generate the waypoints of the optimal path (not necessarily the one desired by the human) to the known position of the goal to which they must take the object. These waypoints generate an attractive force, $\boldsymbol{f_{C,goal}} \in \mathbb{R}^2$, and the weighted sum of this attractive force and the repulsive forces generated by the detected obstacles generates a force representative of the environment, $\boldsymbol{f_{E,C}} \in \mathbb{R}^2$. More details on how to compute this second input to our model can be found in [25], [26].

The third source of information is the force exerted by the human on the transported object and measured by a force sensor on the wrist of the robot, $\boldsymbol{F_{H,C}} \in \mathbb{R}^2$. The fourth source of information corresponds to the linear and angular velocity commands generated by the robot by combining the force exerted by the human with the force representative of the environment. Finally, the fifth input of our model is the distance in modulus and angle to the pre-established goal of the task[1]. In order to make use of and extend the dataset used in our previous work, the last four sources of information mentioned must be normalized to $[-1, 1]$. For this purpose, the following maximum values are considered: 12 N for the modulus of each force, 0.65 m/s for the linear velocity, 1 rad/s for the angular velocity and 7 m for the distance to the goal.

Thus, to obtain both $Y^{force}_{N+1:N+T}$ and $Y^{vel}_{N+1:N+T}$ we will use the last N occupancy maps, $X^{map}_{1:N}$, and the concatenation of the last N values of the other four information entries, $X^{f}_{1:N} = [x^f_1, x^f_2, ..., x^f_N]$ with $x^f_i \in \mathbb{R}^8$. As in [8], we will use the information of the last 2 s to predict the following 1 s ($N = 20$ and $T = 10$ since the system works at 10 Hz).

B. RetNet-based Force/Velocity Predictor Model

In the original article on Retentive Networks (RetNet) [11] they proposed the retention mechanism for modeling sequences by bringing temporal decay to language models:

$$Retention(X) = (QK^T \odot D)V$$
$$D_{n,m} = \begin{cases} \gamma^{n-m} & \text{if } n \geq m \\ 0 & \text{otherwise} \end{cases} \quad (1)$$

with Q, K and V for Queries, Keys and Values; D for causal masking and exponential decay; representing $\odot$ an element-by-element product and n and m the indexes of the selected token and the one with which retention is calculated.

[1]Example of how to calculate $\boldsymbol{F_{Task,C}}$ and the performed experiments: https://youtu.be/Mbxavt78Xvw

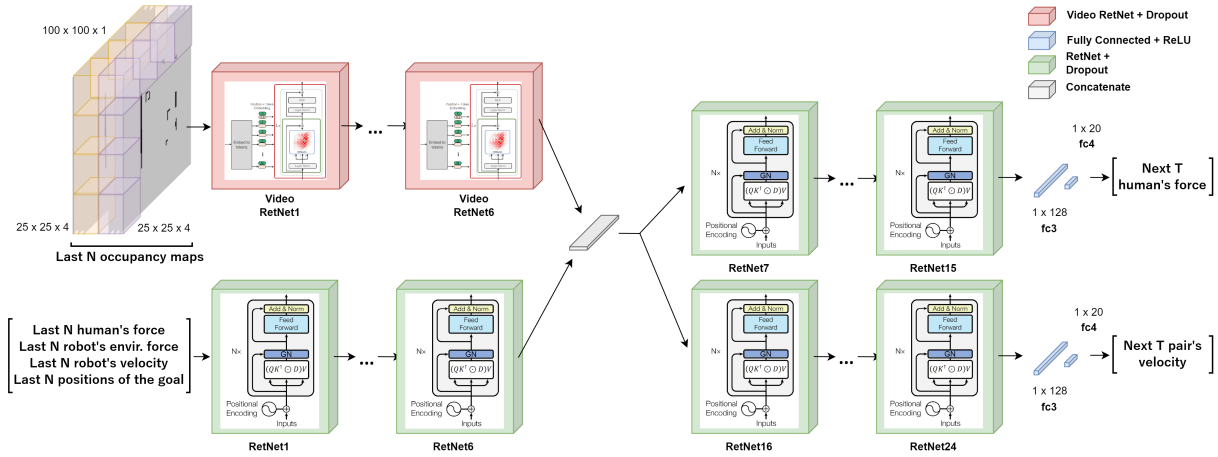

Fig. 2. **Model architecture for RetNet-based force/velocity predictor.** Two input and two output streams in parallel. Occupancy map obtained from LiDAR used as video stream and encoded as tubelets. Our implementation of Video RetNet (3D RetNet) to process them and 1D RetNet to process other inputs. Both streams concatenated and processed to obtain simultaneously a 1 s prediction of next human's force and next human-robot pair's velocity.

a) 1D decay for Retention. b) 2D decay for MaSA. c) 3D decay for 3DMaSA.

Fig. 3. **Different attention mechanisms used with each version of RetNets.** The green square/cube represent the selected token and red squares/cubes represent the relative importance of other tokens with respect to the green one. A - 1D decay for original Retention mechanism [11]. B - Manhathan SefAttention with 2D decay used for generalized RetNets with 2D inputs (images). C - 3D Manhathan SefAttention with 3D decay used for re-generalized RetNets with 3D inputs (videos).

While this mechanism is appropriate for data with causal properties (see Fig. 3 - A), the same is not true when applied to image processing. Because of this, they evolve this mechanism to provide it with bidirectionality and bidimensionality by developing what they call Manhattan Self-Attention (MaSA) [24] (see Fig. 3 - B):

$$\begin{aligned} MaSA(X) &= (Softmax(QK^T) \odot D^{2d})V \\ D^{2d}_{n,m} &= \gamma^{|x_n - x_m| + |y_n - y_m|} \end{aligned} \tag{2}$$

being (x_n, y_n) and (x_m, y_m) the two-dimentional possition of the n-th and m-th considered tokens in the image.

In order to apply this mechanism to videos, where a causal relationship does exist, a generalization must be made which, to the best of our knowledge, has not been considered until this article. To this end, we seek inspiration in [21] and use tubelet embedding, i.e., create non-overlapping, spatio-temporal "tubes" from the input video which are processed as tokens. Subsequently, we generalize MaSA to 3DMaSA in order to process these tokens causally (see Fig. 3 - C):

$$\begin{aligned} 3DMaSA(X) &= (Softmax(QK^T) \odot D^{3d})V \\ D^{3d}_{n,m} &= \begin{cases} \gamma^{|x_n - x_m| + |y_n - y_m| + |z_n - z_m|} & \text{if } z_n \geq z_m \\ 0 & \text{otherwise} \end{cases} \end{aligned} \tag{3}$$

being (x_n, y_n, z_n) and (x_m, y_m, z_m) the three-dimentional possition of the n-th and m-th considered tubelets in the video chunk.

Considering the above, Fig. 2 shows a diagram of our architecture. There are two parallel streams to process $X^{map}_{1:N}$ and $X^{f}_{1:N}$ respectively. $X^{map}_{1:N}$ is sequenced in tubelets containing $L = 4$ consecutive patches of size 25x25 pixels. The choice of these values will be discussed in Section IV. These tubelets are delivered to eight layers of our video-oriented version of RetNets, each of them with $h = 8$ self-attention heads, 128 as the projection dimensionality of queries, keys and values and $p = 0.3$ as the Dropout probability. On the other hand, $X^{f}_{1:N}$ is processed using six layers of vanilla RetNets, each with $h = 8$ retention heads, 64 as the projection dimensionality, 512 for the inner FC layers' dimensionality and 128 for the dimensionality of the sub-layers' outputs. As with the 3D version, Dropout is used for regularization with a probability $p = 0.25$.

The concatenation of both input streams is sent to two output streams to obtain $Y^{force}_{N+1:N+T}$ and $Y^{vel}_{N+1:N+T}$. Both streams consist of nine layers of vanilla RetNets with the same parameters except Dropout $p = 0.3$. This model will be called *RetNet-3D+1D* in the comparisons with other models.

C. Dataset Acquisition and Training

The dataset used in [9] is extended using the samples generated in the real experiments performed in that same work as well as with the experiments carried out in [27] in which our set-up is used to analyze the effect of direct human-robot communication systems. With this, our dataset increases to 18920 sub-sequences, 34% larger than the one previously used. These sub-sequences are the result of dividing each experiment performed in blocks of $N + T$ samples of the five information sources previously indicated and with an overlapping of $(N + T)/2$ samples between sub-sequences. Thus, the first N samples are used by our model to predict the following T human's forces and human-robot pair's velocities. This dataset is divided into the training (90%:

TABLE I

EVOLUTION OF MEAN ERROR AND PERCENTAGE OF CORRECT PREDICTIONS IN TESTSET. VARIABLE Y REPRESENTS FORCE (F) OR VELOCITY (Vel).

Measure		Time [ms]							
		Force (Y = F)				Velocity (Y = Vel)			
		100	300	500	1000	100	300	500	1000
Error Y_x [N or m/s]	CNN+LSTM [8]	0.239	0.250	0.260	0.290	–	–	–	–
	CNN+T [9]	0.205	0.233	0.241	0.254	0.0059	0.0065	0.0073	0.0090
	ViViT+T [9]	**0.190**	0.202	0.211	0.249	**0.0051**	0.0063	0.0071	0.0088
	RetNet-3D+1D	0.192	**0.199**	**0.207**	**0.242**	0.0054	**0.0062**	**0.0069**	**0.0085**
Error Y_y [N or rad/s]	CNN+LSTM [8]	0.121	0.130	0.141	0.163	–	–	–	–
	CNN+T [9]	0.095	0.113	0.117	0.132	0.0040	0.0044	0.0050	0.0064
	ViViT+T [9]	**0.086**	0.096	0.101	0.126	**0.0034**	0.0042	0.0048	0.0062
	RetNet-3D+1D	0.089	**0.095**	**0.100**	**0.125**	0.0036	**0.0041**	**0.0047**	**0.0059**
Error $\lvert\boldsymbol{Y}\rvert < 0.1 \cdot Y_{max}$ & Error $\angle\boldsymbol{Y} < 18°$ [%]	CNN+LSTM [8]	93.7	93.0	92.4	91.2	–	–	–	–
	CNN+T [9]	94.9	94.2	93.9	93.0	98.1	97.6	97.0	96.0
	ViViT+T [9]	**95.6**	94.9	94.4	93.4	**98.4**	97.8	97.2	96.2
	RetNet-3D+1D	95.5	**95.0**	**94.6**	**93.7**	98.3	**97.9**	**97.4**	**96.5**

TABLE II

PERFORMANCE OBTAINED WITH DIFFERENT GRAPHIC CARDS

Model	Frames Per Second (min. / avg. / max.)			
	GTX 1060 Mobile (80 W)	GTX 1660 Ti Desktop	RTX 3060 Mobile (80 W)	RTX 3080 Ti Desktop
CNN+LSTM [8]	13.0 - 13.8 - 14.2	18.0 - 18.8 - 20.1	21.3 - 22.6 - 24.3	54.9 - 58.9 - 62.4
CNN+T [9]	8.98 - 9.79 - 10.2	12.5 - 13.3 - 14.4	15.0 - 16.1 - 17.2	39.9 - 43.0 - 45.8
ViViT+T [9]	6.72 - 7.26 - 7.82	9.07 - 9.84 - 10.6	10.7 - 11.9 - 12.9	30.2 - 32.3 - 34.5
RetNet-3D+1D	8.54 - 9.21 - 10.0	11.7 - 12.4 - 13.4	14.4 - 15.5 - 16.6	40.1 - 42.9 - 45.6

17028 sub-sequences), validation (5%: 946 sub-sequences) and testing (5%: 946 sub-sequences) datasets.

The model is coded using the Keras 3.0 API, which is compatible with both TensorFlow and PyTorch. In this way, both the previous models and the new one are retrained using the extended dataset in order to compare their performance. The optimizer used is Adam with its default parameters except for the learning rate, which is used $lr = 5 \times 10^{-4}$. Learning rate decay is also used with a decay factor of 0.96 up to $lr_{min} = 3 \times 10^{-5}$. Early stopping is added to avoid overfitting. The maximum number of epochs is set at 100 although no model exceeded epoch 91 due to early stopping. An NVIDIA RTX 2080 Ti graphics card was used, training for 110-170 minutes depending on the model.

IV. RESULTS

First, we must check the performance of our predictor compared to the predictors designed in our previous work both in predicting the next force to be exerted by the human and the velocity of the human-robot pair. To do so, we will perform multiple tests on the testset split. Once this is done, we take the best predictor among the previously designed ones and our new predictor and test its performance in real experiments. To do this, we record 15 new experiments performed by volunteers who had not previously done this task in previous rounds of experiments, so that they may have different preferences than those present in the dataset. These experiments are performed without executing any of the predictors so that they do not condition the behavior of the robot, in this case the IVO robot [28]. Offline, the data recordings of the experiments are played back running each predictor encapsulated in a ROS (Robot Operating System) node so that its real performance can be checked. All the experiments reported in this work have been performed after getting the approval of the ethics committee of the Universitat Politècnica de Catalunya (UPC) in accordance with all the regulations and relevant guidelines (ID: 2023.05).

A. Force/Velocity Predictor Performance in Dataset

To check the performance of our predictor, we use the metrics outlined in [8]. Thus, to evaluate the force prediction, we calculate the absolute error made on each Cartesian axis by comparing the prediction with its actual value. Likewise, to evaluate the velocity prediction, we calculate the absolute error between the linear and angular velocity prediction and their respective real values. In addition, for both predictions, we also calculate the percentage of samples that show an error in modulus and angle of less than 10% (1.2 N for the force and 0.065 m/s and 0.1 rad/s for the velocity).

Table I shows the evolution of the above metrics. It is worth mentioning that the results obtained by the previous predictors, CNN+LSTM, CNN+T and ViViT+T differ

TABLE III

ABLATION STUDY WITH RETNET-3D+1D REMOVING EACH INPUT. VARIABLE Y REPRESENTS FORCE (F) OR VELOCITY (Vel).

Measure		Time [ms]							
		Force ($Y = F$)				Velocity ($Y = Vel$)			
		100	300	500	1000	100	300	500	1000
Error $\|\boldsymbol{Y}\| < 0.1 \cdot Y_{max}$ & Error $\angle\boldsymbol{Y} < 18°$ [%]	Without ocupancy map	86.0	83.2	80.2	75.1	93.7	91.2	87.9	84.4
	Without env. force	93.4	91.1	88.9	85.2	96.5	95.4	93.3	90.4
	Without human's force	90.4	88.6	85.7	80.9	96.3	95.1	93.0	90.0
	Without robot's velocity	93.5	90.9	88.3	84.2	95.7	94.4	92.1	88.7
	Without goal position	94.8	94.1	93.3	92.1	97.7	97.0	96.2	95.0

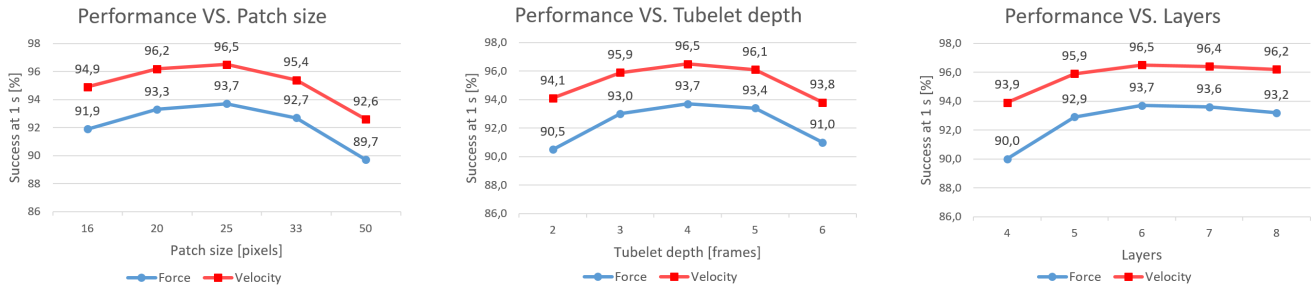

Fig. 4. **Evolution of force (in blue) and velocity (in red) prediction accuracy at $1\ s$ using diferent model hyperparameters in the Video RetNet stream.** *Left* - Performance variation by changing the occupancy map patch size. *Middle* - Performance variation by changing the tubelet depth. *Right* - Performance variation by changing the number of Video RetNet layers.

slightly from those shown in [9] because all models have been trained with a larger dataset over more epochs. First, ViViT+T outperforms CNN+T, a result that did not occur in [9]. It confirms the known issue in the literature that ViT (and thus ViViT) can outperform CNNs in image recognition tasks if trained with sufficiently large datasets [29], [30]. Second, RetNet-3D+1D manages to outperform all other predictors from about the first 200 ms. However, this observed improvement between RetNet-3D+1D over ViViT+T is inferior to that achieved with ViViT+T over CNN+LSTM. This could be indicative that we are reaching the limit of the predictive capabilities of these variables since the force exerted by the human is intimately related to their intention and this can change suddenly.

On the other hand, RetNets do not stand out over Transformers so much for a better performance in the task for which they are used but rather for a more efficient use of computational resources, especially memory [11]. This is reflected in the Table II in which the Frames Per Second or FPS (calculated as the inverse of the inference time or time it takes to execute the line of code in which a new prediction is requested after receiving a new set of input data) of each model executed on different graphics cards is checked. The performance improvement observed over ViViT+T makes the use of RetNet-3D+1D to make sense, especially when using lower-end hardware. It should be noted that since they are graphics cards from different generations, it is not possible to use the same drivers version in all of them. Instead, we used the most updated graphics driver available for each of them as well as the most recent CUDA compatible version.

Focusing on the RetNet-3D+1D model, the testset can be used to perform an ablation test to know the relative importance of each input. To do this, predictions are made by hiding the different inputs one at a time and the percentage of samples with an error of less than 10% is calculated. The Table III shows the results. As expected, the most relevant factor is the occupancy map because without it the full processing power of one of the model's input streams is lost. Its absence causes a drop in performance of up to 18.6% in force prediction and up to 12.1% in velocity prediction. The next most relevant factors are the force exerted by the human previously causing drops of up to 12.8% and up to 6.5% respectively, and the speed followed by the human-robot pair with drops of up to 9.5% and 7.8%. Note that the force exerted turns out to be less determinant in predicting the speed of the pair and vice versa. Finally, the least determinant factors are the virtual force generated to represent the environment with drops of up to 8.5% and 6.1%, and the distance in modulus and angle to the goal with drops of 1.6% and 1.5%. This last and relatively small drop in performance by eliminating the goal position offers the possibility of using our predictor in tasks where the goal is not pre-set but can be chosen by the human on the fly, although its testing is outside the scope of this article.

Finally, and considering that the most significant contribution of this article is the generalization of RetNets to be used for video processing and its application to the task of human-robot collaborative transportation, Fig. 4 shows how the performance of the model varies both when predicting the force to be exerted and the speed of the pair by varying

TABLE IV

MEAN ERROR AND PERCENTAGE OF CORRECT PREDICTIONS IN REAL EXPERIMENTS. VARIABLE Y REPRESENTS FORCE (F) OR VELOCITY (Vel).

Measure		Time [ms]							
		Force ($Y = F$)				Velocity ($Y = Vel$)			
		100	300	500	1000	100	300	500	1000
Error Y_x [N or m/s]	ViViT+T [9]	**0.240**	0.253	0.265	0.312	**0.0070**	0.0084	0.0098	0.0116
	RetNet-3D+1D	0.242	**0.252**	**0.261**	**0.298**	0.0072	**0.0083**	**0.0095**	**0.0113**
	RetNet-3D+1D (w/o goal)	0.251	0.264	0.295	0.330	0.0086	0.0101	0.0120	0.0148
Error Y_y [N or rad/s]	ViViT+T [9]	**0.115**	0.129	0.145	0.173	**0.0048**	0.0057	0.0077	0.0100
	RetNet-3D+1D	0.117	**0.129**	**0.142**	**0.169**	0.0050	**0.0056**	**0.0075**	**0.0095**
	RetNet-3D+1D (w/o goal)	0.128	0.144	0.169	0.179	0.0060	0.0080	0.0102	0.0121
Error $\lvert\boldsymbol{Y}\rvert < 0.1 \cdot Y_{max}$ & Error $\angle\boldsymbol{Y} < 18°$ [%]	ViViT+T [9]	**94.0**	93.1	92.2	90.8	**97.2**	96.5	95.8	94.7
	RetNet-3D+1D	93.9	**93.1**	**92.3**	**91.0**	97.1	**96.6**	**95.9**	**95.0**
	RetNet-3D+1D (w/o goal)	93.2	92.2	91.0	89.6	96.4	95.6	94.6	93.4

TABLE V

COMPARISON OF MEAN ERROR ESTIMATING HUMAN TRAJECTORY WITH DIFFERENT MODELS. * MARKS VALUES OBTAINED BY INTERPOLATION FROM LAPLAZA EL AL. [3].

Model	L2 [m]	
	500 ms	1000 ms
Martinez el al. [31]	0.159*	0.317*
Mao et al. [32]	0.081*	0.161*
Laplaza et al. [3]	0.072*	0.142*
2nd order polynomial	0.123	0.277
CNN+LSTM [8]	0.095	0.202
ViViT+T [9]	0.063	0.142
RetNet-3D+1D	**0.061**	**0.139**
RetNet-3D+1D (w/o goal)	0.084	0.171

the main parameters of the architecture. Although these results are considerably task-specific, we believe that they may be useful to the reader when tuning their own model based on this architecture. In this case, it can be observed that the maximum performance is achieved for a patch size equivalent to one-sixteenth of the input image size: 25x25 pixels versus 100x100. On the other hand, the best result is obtained with a tubelet depth of one fifth of the number of images composing the sequence: 4 versus 20 (2 s at 10 Hz). As for the number of layers, the performance could still increase above 6 layers if a larger dataset were available.

B. Predictor Performance in Real Experiments

Although the testset split by definition is composed of samples that are not used during training, they belong to the same distribution as they are obtained after shuffling and splitting all the samples obtained from the experiments performed so far. That is why to check the real performance of our predictor we use 15 new real experiments not present in the dataset. Table IV shows the results. Analyzing the RetNet-3D+1D model, there is a drop in performance of up to 1.7% in force prediction and up to 1.5% in velocity prediction. This model still performs better than ViViT+T in both predictions at 200 ms and above. We also include the performance offered by the RetNet-3D+1D model in the case of not giving it the goal position to get a rough idea of the effect that the absence of this variable would have in real experiments.

As previously mentioned, this prediction of the speed that the human-robot pair will follow can be integrated to obtain an estimate of the trajectory they will follow. Table V shows a comparison of the error made when estimating the trajectory with different models. While [3], [31], [32] are predictors of human movement applied to other tasks such as handover and therefore cannot be used to make a fair comparison, they do serve to give context to the predictive capabilities of our model. As it can be seen, RetNet-3D+1D achieves the lowest error beating ViViT+T. On the other hand, the version of this model that does not receive goal position information shows a significant increase in the error, although it improves the result offered by our original predictor [8].

V. CONCLUSIONS AND FUTURE WORK

In this work, a generalization of the recent RetNet architecture has been presented so that it can be used to process video as input. Results are provided on the effect of varying the most important parameters in this architecture. Although these results have been obtained in a specific task, we believe that they can be useful to design other models applied to other tasks based on this architecture.

Secondly, we provide a new force/velocity predictor applied to the task of collaborative transport of objects between human and robot that obtains a higher accuracy in both predictions and that makes a more efficient use of resources reaching improvements of up to 32.8% in the FPS obtained, which allows us to use our new predictor in real time using lower-end hardware. Through real experiments, it proves to

be the best predictor reaching acceptable error rates 91.0% of the time for the prediction of the force to be exerted by the human in 1 s and over 95.0% for the speed of the human-robot pair. Additionally, it also shows the best capabilities if used to generate an estimate of the trajectory that both agents will follow.

For future work, through an ablation study we have found that the elimination of one of the information inputs, such as the position of the until now pre-set goal of the task, does not generate a high performance drop. This opens the door to generalize our predictor so that the task goal can be chosen on the fly by the human. A new round of experiments, as well as possible changes in the architecture, will be necessary to test this possibility. In addition, other information inputs such as the human's gaze could be taken into account to overcome the performance limit that we seem to have reached with the inputs considered so far.

REFERENCES

[1] L. Kunze, N. Hawes, T. Duckett, M. Hanheide, and T. Krajník, "Artificial Intelligence for Long-Term Robot Autonomy: A Survey," *IEEE Robotics and Automation Letters*, vol. 3, no. 4, pp. 4023–4030, 2018.

[2] Y. Ma, Z. Wang, H. Yang, and L. Yang, "Artificial Intelligence Applications in the Development of Autonomous Vehicles: A Survey," *IEEE/CAA Journal of Automatica Sinica*, vol. 7, no. 2, pp. 315–329, 2020.

[3] J. Laplaza, F. Moreno-Noguer, and A. Sanfeliu, "Context and Intention aware 3D Human Body Motion Prediction using an Attention Deep Learning model in Handover Tasks." IEEE, 2022, pp. 4743–4748.

[4] J. E. Domínguez-Vidal, I. J. Torres-Rodríguez, A. Garrell, and A. Sanfeliu, "User-Friendly Smartphone Interface to Share Knowledge in Human-Robot Collaborative Search Tasks," in *30th IEEE International Conference on Robot and Human Interactive Communication (RO-MAN)*, 2021, pp. 913–918.

[5] M. Dalmasso, J. E. Domínguez-Vidal, I. J. Torres-Rodríguez, A. Garrell, and A. Sanfeliu, "Shared Task Representation for Human-Robot Collaborative Navigation: The Collaborative Search Case," *International Journal of Social Robotics*, 2023.

[6] Y. Cheng and M. Tomizuka, "Long-term trajectory prediction of the human hand and duration estimation of the human action," *IEEE Robotics and Automation Letters*, vol. 7, no. 1, pp. 247–254, 2021.

[7] Z. Zhang, G. Peng, W. Wang, Y. Chen, Y. Jia, and S. Liu, "Prediction-based human-robot collaboration in assembly tasks using a learning from demonstration model," *Sensors*, vol. 22, no. 11, p. 4279, 2022.

[8] J. E. Domínguez-Vidal and A. Sanfeliu, "Improving Human-Robot Interaction Effectiveness in Human-Robot Collaborative Object Transportation using Force Prediction," in *2023 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2023, pp. 7839–7845.

[9] J. E. Domínguez-Vidal and A. Sanfeliu, "Exploring Transformers and Visual Transformers for Force Prediction in Human-Robot Collaborative Transportation Tasks," in *2024 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2024, p. to appear.

[10] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," *Advances in neural information processing systems*, vol. 30, 2017.

[11] Y. Sun, L. Dong, S. Huang, S. Ma, Y. Xia, J. Xue, J. Wang, and F. Wei, "Retentive Network: A Successor to Transformer for Large Language Models," 2023.

[12] A. Bussy, P. Gergondet, A. Kheddar, F. Keith, and A. Crosnier, "Proactive behavior of a humanoid robot in a haptic transportation task with a human partner," in *2012 IEEE RO-MAN: The 21st IEEE International Symposium on Robot and Human Interactive Communication*. IEEE, 2012, pp. 962–967.

[13] S. Tarbouriech, B. Navarro, P. Fraisse, A. Crosnier, A. Cherubini, and D. Sallé, "Admittance control for collaborative dual-arm manipulation," in *2019 19th International Conference on Advanced Robotics (ICAR)*. IEEE, 2019, pp. 198–204.

[14] D. J. Agravante, A. Cherubini, A. Bussy, P. Gergondet, and A. Kheddar, "Collaborative human-humanoid carrying using vision and haptic sensing," in *2014 IEEE international conference on robotics and automation (ICRA)*. IEEE, 2014, pp. 607–612.

[15] X. Yu, B. Li, W. He, Y. Feng, L. Cheng, and C. Silvestre, "Adaptive-constrained impedance control for human–robot co-transportation," *IEEE transactions on cybernetics*, vol. 52, no. 12, pp. 13 237–13 249, 2021.

[16] X. Yu, W. He, Y. Li, C. Xue, J. Li, J. Zou, and C. Yang, "Bayesian estimation of human impedance and motion intention for human–robot collaboration," *IEEE transactions on cybernetics*, vol. 51, no. 4, pp. 1822–1834, 2019.

[17] C. N. Mavridis, K. Alevizos, C. P. Bechlioulis, and K. J. Kyriakopoulos, "Human-robot collaboration based on robust motion intention estimation with prescribed performance," in *2018 European Control Conference (ECC)*. IEEE, 2018, pp. 249–254.

[18] L. Roveda, J. Maskani, P. Franceschi, A. Abdi, F. Braghin, L. Molinari Tosatti, and N. Pedrocchi, "Model-Based Reinforcement Learning Variable Impedance Control for Human-Robot Collaboration," *Journal of Intelligent & Robotic Systems*, vol. 100, no. 2, pp. 417–433, 2020.

[19] E. Gribovskaya, A. Kheddar, and A. Billard, "Motion learning and adaptive impedance for robot control during physical interaction with humans," in *2011 IEEE International Conference on Robotics and Automation*. IEEE, 2011, pp. 4326–4332.

[20] A. Al-Yacoub, Y. Zhao, W. Eaton, Y. M. Goh, and N. Lohse, "Improving human robot collaboration through force/torque based learning for object manipulation," *Robotics and Computer-Integrated Manufacturing*, vol. 69, p. 102111, 2021.

[21] A. Arnab, M. Dehghani, G. Heigold, C. Sun, M. Lučić, and C. Schmid, "Vivit: A video vision transformer," in *Proceedings of the IEEE/CVF international conference on computer vision*, 2021, pp. 6836–6846.

[22] A. Katharopoulos, A. Vyas, N. Pappas, and F. Fleuret, "Transformers are RNNs: Fast Autoregressive Transformers with Linear Attention," in *International conference on machine learning*. PMLR, 2020, pp. 5156–5165.

[23] B. Peng, E. Alcaide, Q. Anthony, A. Albalak, S. Arcadinho, H. Cao, X. Cheng, M. Chung, M. Grella, K. K. GV *et al.*, "RWKV: Reinventing RNNs for the Transformer Era," *arXiv preprint arXiv:2305.13048*, 2023.

[24] Q. Fan, H. Huang, M. Chen, H. Liu, and R. He, "RMT: Retentive Networks Meet Vision Transformers," *arXiv preprint arXiv:2309.11523*, 2023.

[25] J. E. Domínguez-Vidal, N. Rodríguez, and A. Sanfeliu, "Perception-Intention-Action Cycle as a Human Acceptable Way for Improving Human-Robot Collaborative Tasks," in *Proceedings of the 2023 ACM/IEEE International Conference on Human-Robot Interaction*, 2023, p. 567–571.

[26] J. E. Domínguez-Vidal, N. Rodríguez, and A. Sanfeliu, "Perception-Intention-Action Cycle in Human-Robot Collaborative Tasks: the Collaborative Lightweight Object Transportation Use-Case," *International Journal of Social Robotics*, p. to appear, 2024.

[27] J. E. Domínguez-Vidal and A. Sanfeliu, "Voice Command Recognition for Explicit Intent Elicitation in Collaborative Object Transportation Tasks: a ROS-based Implementation," in *Companion of the 2024 ACM/IEEE International Conference on Human-Robot Interaction*, 2024, p. to appear.

[28] J. Laplaza, N. Rodríguez, J. E. Domínguez-Vidal, F. Herrero, S. Hernández, A. López, A. Sanfeliu, and A. Garrell, "IVO Robot: A New Social Robot for Human-Robot Collaboration," in *Proceedings of the 2022 ACM/IEEE International Conference on Human-Robot Interaction*. IEEE, 2022, p. 860–864.

[29] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly *et al.*, "An image is worth 16x16 words: Transformers for image recognition at scale," *arXiv preprint arXiv:2010.11929*, 2020.

[30] L. Deininger, B. Stimpel, A. Yuce, S. Abbasi-Sureshjani, S. Schönenberger, P. Ocampo, K. Korski, and F. Gaire, "A comparative study between vision transformers and cnns in digital pathology," *arXiv preprint arXiv:2206.00389*, 2022.

[31] J. Martinez, M. J. Black, and J. Romero, "On human motion prediction using recurrent neural networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 2891–2900.

[32] W. Mao, M. Liu, and M. Salzmann, "History repeats itself: Human motion prediction via motion attention," in *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XIV 16*. Springer, 2020, pp. 474–489.