



**HAL**  
open science

# Real Time Dynamic System Stochastic Identification in Video Capture for Data Compression, Image Interpolation, Prediction, and Augmented Reality

Adrien Goeller, Jean-Luc Dion, Thierry Soriano, Bernard Roux

## ► To cite this version:

Adrien Goeller, Jean-Luc Dion, Thierry Soriano, Bernard Roux. Real Time Dynamic System Stochastic Identification in Video Capture for Data Compression, Image Interpolation, Prediction, and Augmented Reality. ASME 2015 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference, Aug 2015, Boston, United States. pp.V008T13A054–V008T13A054, 10.1115/DETC2015-47398 . hal-01723798

**HAL Id: hal-01723798**

**<https://univ-tln.hal.science/hal-01723798>**

Submitted on 25 Nov 2021

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

# REAL TIME DYNAMIC SYSTEM STOCHASTIC IDENTIFICATION IN VIDEO CAPTURE FOR DATA COMPRESSION, IMAGE INTERPOLATION, PREDICTION, AND AUGMENTED REALITY

**Adrien GOELLER**  
SUPMECA - EA 2336  
Saint-Ouen, France

**Jean-Luc DION**  
SUPMECA - EA 2336  
Saint-Ouen, France

**Thierry SORIANO**  
SUPMECA  
La Garde, France

**Bernard ROUX**  
Vannier-Kinoptik  
Antony, France

## ABSTRACT

In computer vision, cameras more and more accurate, fast, 3D featured are used. These still evolutions generate more data, which is an issue for users to store it with standard compression for example for recording proof in case of products manufacture defective.

The aim of this work is to develop a specific solution adapted for vision systems which have a known scenario and can be described by dynamic models. In this framework, Kalman filters are used for data compression, observable variable prediction, and augmented reality. The developed concepts are tested with a scenario of a ruler on a table. The experiment aims to check the data compression level, the estimation of the friction forces coefficient of the ruler and the prediction of the stop position.

## KEYWORDS

Stochastic identification, Kalman filters, Augmented reality, Prediction.

## 1 INTRODUCTION

### 1.1 INDUSTRIAL CONTEXT

Being a European manufacturer is not as easy as twenty years ago [1]. The rise of the Asian concurrency in the last two decades has brought the necessity in Europe to find another added value. European manufacturers had to deal with different financial constraints, and need to take advantage from their knowledge and their position on the global market. Quality assurance is one of the ways to raise the added value and capitalise from their position. [2]

It is in this framework vision machines take action. In order to have a better manufacturing in terms of traceability, speed, and quality, vision is overwhelming more and more industrial domains. Currently, vision machines are able to measure a lot amount of parameters in order to level the controls. There is a lot of manufacturers who wants a customized control on a large number of parameters in several part of the production chain, and merge it on a database in order to keep tracks in case of a defective product. The immediate consequence is a large amount of data, several cameras in order to have many points of observations for inspection and for 3D, and limitations on speed. Besides, some hidden physical parameters of manufactured object are not accessible because they are not visual parameters: It is currently compensated by

the high use of augmented reality, merging visual data with data coming from other sensors type.

The method introduced in this article is promising to enable vision machine dealing with all the points mentioned before by improving the comprehension of the object by the computer. Based on a physical model of the object seen, this algorithm enables the machine to understand the behaviour of the scene and to extract some hidden physical parameters directly from observation. A hidden parameter means that the parameter cannot be measured without specific instrumentation except a camera.

The results could be interesting in the second industrial context of high speed videos. This type of cameras is used in order to measure deformation, speed, or efforts in the example of crash tests, or to understand some high speed phenomenon. The method introduced in this article is promising to add information holding on the same physical model. In addition, the model is allowing to produce interpolated frames as soon as there is no aliasing [3] regarding to the ratio frame acquisition/object speed.

## 1.2 PAPER PURPOSE

This paper deals with a subject parted in several blocks. The main purpose of this paper is to develop a method to mix data extracted from the video of a dynamic scene with a parametric mechanical model describing the video. The merging and fitting process between data and parameters model is processed using Kalman Filters. Even if the method presented here are not time related yet, the aim in the industrial context is to reach a real time application.

The paper purpose is double: on one hand, to develop the concept theory based on Kalman filters and object tracking, on the other hand, to prove its feasibility with a simple 2D example on a video scene. A qualitative and quantitative analyse of the results shows the kind of results expected on the different fields of improvement, which are prediction, augmented reality, high compression, and frame interpolation.

## 1.3 IMAGE/VIDEO PROCESSING REFERENCES

In order to extract data from video, an algorithm needs to be developed but the method has to be chosen regarding to the execution speed of the method.

In this way, different method from image or video processing could be applied. The related work used to understand the different existing methods, their efficiency, their precision, and the computational load such as Object tracking [4], or background detection [5].

However, this image processing part is not the purpose of this article which deals only with a numerical simulation. The experimental part with the image processing block is going to be implemented in a second time.

## 1.4 KALMAN FILTERS

The present work aims to identify variables and parameters by using Unscented Kalman Filters. [6]

In the field of dynamics, several works have already used Kalman filters for identification, detection, health monitoring or tracking order. In identification and detection processes, Kalman Filters are performed for the localization of cracks on rotating machines [7], for stiffness identification [8] or vibration force estimation [9], [10]. Some health monitoring methods for structures and rotating machines are based on Kalman Filters with varying Auto Regressive identification parameters [11], [12], [13], [14], [15]. Kalman Filters are also used in active vibration control [16], [17] [18] with real-time algorithms and non-stationary signals on smart structures.

This work aims to track the trajectory of an object on a video. This work has to be based on ballistic tracking. Some comparative articles between different tracking algorithms, highlighting specifically the computational load, confirming the choice made here to use Extended or Unscented Kalman Filter [19]. Other work in ballistic tracking field using Kalman shows the possibility to use sophisticated model with a Kalman Filter [20].

## 2 EXPERIMENTAL USE CASE

To illustrate properly all the four improvement consequence of the method, a simple use case has been developed. This use case would have been a dynamic scene easily achievable, but it had to deal with physical parameters which are not visual and had to be described by mechanical equations. The example of a ruler sliding upon a table is satisfying each point previously mentioned: on the mechanical side the model is not quite simple, there is different hidden physical parameter which are the friction force and the position of the two contact points with the table, on the realization side, the staging is easy and the variation of the friction force or the trajectory is easy.

### 2.1 NUMERICAL TEST BENCH

In order to have a perfect comparison on the measures performed with the method, and the real parameters measured, a numerical test bench has to be developed to escape every bias due to another comparison measure, but to escape any propagation error in the image acquisition chain.

This test bench allowed following a block working way. As a first step, only the value of the parameters measured further by the video are used. There is no image processing in this first part, enable to compare the initial set of parameters with the same set assessed by the method. As a second step, a video is generated with the previous values, and it enables to set up the image processing part.

The model chosen to simulate the behaviour of the ruler is a double sliding contact point, A and B in the Diagram 1: ruler consideration for study, which models the ruler resistance by solid friction.  $T_A$  and  $T_B$  are the friction forces applied on A and B.

The position of A and B could be variable in order to model some rough variation on a real table. The position of the two sliding contacts enables to model a ruler not perfectly flat. The equation system below is solved by an explicit ODE method to generate the batch of value for the next step.

$$\begin{cases} m\ddot{x} = T_{Ax} + T_{Bx} \\ m\ddot{y} = T_{Ay} + T_{By} \\ \bar{I}\ddot{\alpha} = L_1(T_{Ax}\cdot\sin(\alpha) - T_{Ay}\cdot\cos(\alpha)) \\ -L_2(T_{Bx}\cdot\sin(\alpha) - T_{By}\cdot\cos(\alpha)) \end{cases}$$

In this use case, the image processing has been chosen to assess the initial position or speed of the ruler, but it could have been part of the unknowns.

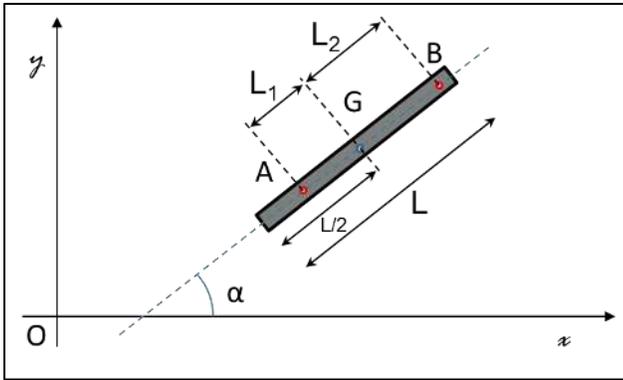


Diagram 1: Ruler consideration for the study

## 2.2 VIDEO OUTPUT

In order to perform the image processing block, a video output is implemented in output of the ruler event simulator. The goal of this video is to enable to see the performances of the image processing block, and to allow to test the complete numerical part of the process. The visual result can be impressive [21].

The ruler is simulated by a rectangle between the two edge points calculated with  $(x_G, y_G, \alpha, L)$ .

This video output saves the image processing block from any noise reflection of light, which will be an issue on real videos. A legend of the different information on the video is given in Diagram 2 and examples of the output are in Annex E and F.

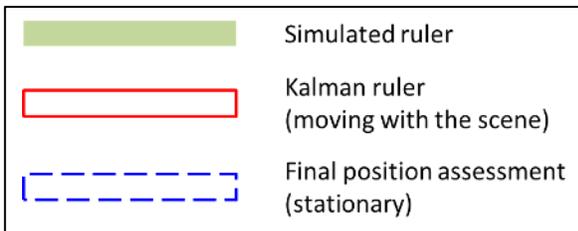


Diagram 2: legend of the video output

In order to understand, the Diagram 3 is giving a simplified view on the simulation process.

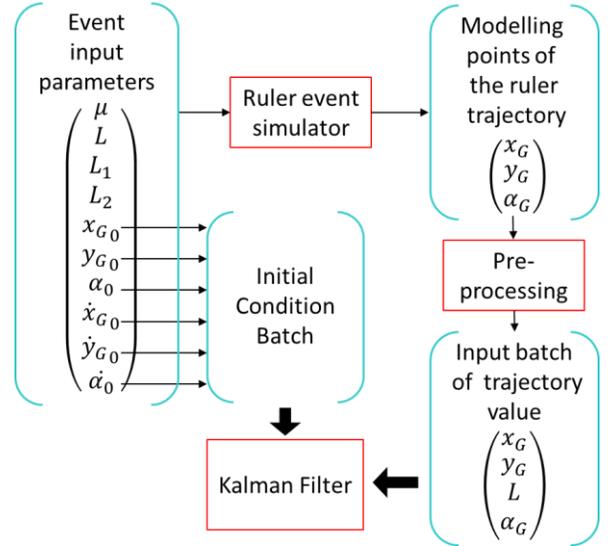


Diagram 3: Simplified view of the simulation process

## 3 STOCHASTIC IDENTIFICATION PROCESS: THE UNSCENTED KALMAN FILTER

### 3.1 KALMAN FILTERS

Kalman filtering refers to a family of algorithms that tracks the temporal evolution of a dynamic model based on noised measurements:

$$\begin{cases} X_{k+1} = f(X_k, w_k) \\ Z_{k+1} = h(X_{k+1}, v_{k+1}) \end{cases}$$

This evolution is described here in the discrete time domain. The first equation of the system above is the propagation equation whereas the second is the observation equation.  $X_k$  is the state vector at the step  $k$ ,  $f(\cdot)$  is the model function,  $w_k$  the process noise assumed to be Gaussian  $N(0, Q)$ , and  $Z_k$  is the observation vector,  $h(\cdot)$  the observation function and  $v_k$  the observation noise assumed to be Gaussian  $N(0, R)$ .

The Kalman filter is an efficient algorithm able to predict the future state vector using its means and covariances. [22] It is based on a model of the system described by the function  $f(\cdot)$  compared to the observation  $Z_k$  with the observation function  $h(\cdot)$ . The Kalman filter is based on two major hypotheses: the model function and the observation function are linear, and the variables considered in the system are Gaussian. Due to the linear hypotheses,  $f(\cdot)$  is the F matrix and  $h(\cdot)$  the H matrix in the following.

An approach of Kalman algorithm is to separate it in two phases. The predict phase gives an a priori estimate of the state and covariance based on previous time step  $t_k$ :

$$\hat{X}_{k+1|k} = F_{k|k}\hat{X}_{k|k}$$

$$\hat{P}_{k+1|k}^{XX} = F_{k|k} \hat{P}_{k|k}^{XX} F_{k|k}^T + Q$$

The update phase corrects the deviation of these estimations based on new observations at time stem  $t_{k+1}$ . It is composed by an innovation phase:

$$\hat{Y}_{k+1|k} = Z_{k+1} - H \hat{X}_{k+1|k}$$

$$\hat{P}_{k+1|k}^{YY} = H \hat{P}_{k|k}^{YY} H^T + R$$

Then the calculus of the Kalman gain:

$$K_{k+1} = \hat{P}_{k+1|k}^{XX} H^T (\hat{P}_{k+1|k}^{YY})^{-1}$$

Finally the update of the state and covariance:

$$\hat{X}_{k+1|k+1} = \hat{X}_{k+1|k} + K_{k+1} \hat{Y}_{k+1|k}$$

$$\hat{P}_{k+1|k+1}^{XX} = (I - K_{k+1} H) \hat{P}_{k+1|k}^{XX}$$

### 3.2 UNSCENTED FALMAN FILTER

Kalman has been derived in a lot of different forms in order to pass through the problem of non-linear cases. Among those methods, the Unscented Kalman filter method (called UKF) has been chosen to deal with the two issues of our case: the mechanical model is highly nonlinear, and it is described by unsolved differential equations.

Unscented Kalman filter has a totally different approach compared to others filters as Extended Kalman filters. [6] The Extended version proposed to linearize using a first order development of the nonlinear function. The Unscented version proposes to use the probability distribution to approximate the nonlinear function.

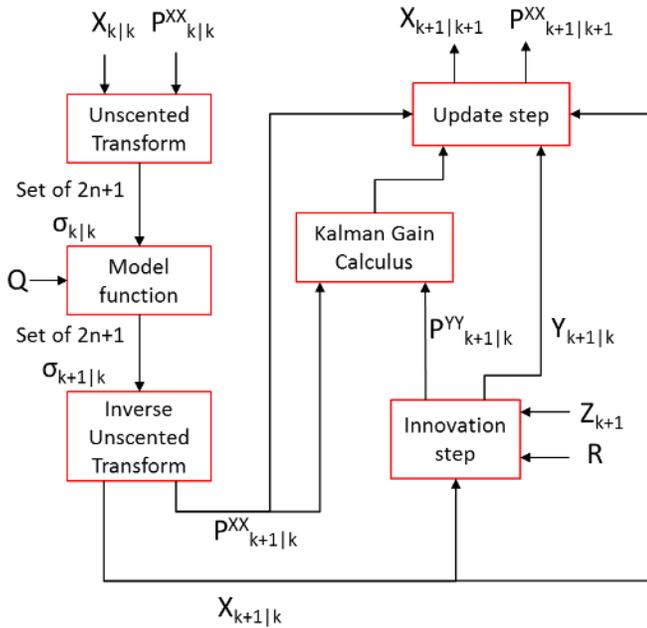


Diagram 4: Algorithm of the Unscented Falman Filter

The unscented transform enables the propagation of the state vector through the model function  $f(\cdot)$  or the observation function  $h(\cdot)$ . In this way, a deterministic sampling technique is

used to catch the state probability distribution and the two first moments. The sampling set is composed of sigma points and those sigma points are propagated through the nonlinear function. The minimal number of points is  $n+1$  with  $n$  the size of the state vector. In our case, it has been chosen  $2n+1$  sigma points. After that, the inverse unscented transformation is used to obtain the state vector and the covariance from the sigma points. This method can be repeated for the innovation phase whether the observation function is nonlinear too.

After the unscented step, the standard Kalman filter can be applied from the update phase adding the innovation phase if the function is linear. A condensed algorithm is given on Diagram 4.

### 3.3 STATE SYSTEM CHOICE

The purpose of the article is to make a proof of working. In this scope, referring to the Diagram 3, the different values to be assessed to describe the ruler motion are the input batch from the trajectory  $(x_G, y_G, \alpha, L)$ . However, the model is using the translation and rotation speed as intermediate variables. These speeds have been added into the state vector in order to have a smoother value.

The other value needed to be assessed is the friction coefficient  $\mu$  and the position of the virtual points  $L_1$  and  $L_2$ .

The motion equations are quickly complicating to a nonlinear form due to the intervention of sinus/cosinus for the intervention of  $\alpha$ . Moreover the differential system is a coupled one, making an explicit solution impossible. All these points lead to choose the Unscented Kalman filter with a Runge-Kutta 4 solving method.

The adopted approach is to make an observation matrix simple and linear and transfer all the nonlinearities to the model function.

The observation vector at discrete time  $i$  is

$$Z_i = \begin{pmatrix} x_G \\ y_G \\ L \\ \alpha \end{pmatrix}$$

The state vector at discrete time  $i$  is:

$$X_i = \begin{pmatrix} x_G \\ y_G \\ L \\ \alpha \\ V_x \\ V_y \\ V_\alpha \\ L_1 \\ L_2 \\ \mu \end{pmatrix}$$

The propagation matrix is linear and has the form:

$$H_i = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$

## 4 SIMULATION RESULTS DISCUSSIONS & PERSPECTIVES

### 4.1 RESULTS

In order to analyze the results correctly, some scenarios have been chosen (see Annex A). These scenarios have been selected in order to test the model varying the different input. One version with an important initial translation speed, a second one with an important initial rotation initial speed, and a third one with both. These variations are preponderant in the parameters identification because the parameters in the models are more or less enhanced depending on the type of movement simulated.

The presented results were simulated directly from the original data used to generate the simulated video, which means the input data is not biased neither by the video generation nor the object tracking.

The results are presented in the Annex B, C, and D. They have been generated using the same initialization covariance, but with initial conditions on position and speed close to real, that is less than 20% of error. However, the parameters to assess have the same set values in all the simulation: 0.05 for the friction forces and 0.25 m for the contact point distance.

The results are very impressive for the observed values. The position, angle and size of the ruler are very precise and even more precise than the noisy information provided to the filter. The friction coefficient is always and quickly assessed with a static error of about 10%. The contact point distance is assessed but this is a very noisy signal.

### 4.2 DATA COMPRESSION

The term of compression used here has been chosen for characterising the initial video size/size of the different element needed to generate the video.

Here, the video size is 132 MB because not compressed. In order to generate the video, the mechanical model (10 kB), the state vector matrix (7 kB), and a generator are needed (10 kB) with the background image and the object image which means about 3 MB for a HD quality image.

Thus this compression ratio is 2-3% maximum. This is important to understand that here this is not the precision on the entire image which is expected, but on the tracked values in the state vector.

### 4.3 IMAGE INTERPOLATION

The approach of the image interpolation is totally different than image processing. Here, pixel based methods are not mentioned. The confidence is on the model, and the interpolated frames are generated regarding only the acquisition frequency: the model is following the nonlinear behavior of the object, thus the interpolated frames are as precise as the model precision.

### 4.4 PREDICTION

The ruler motion equations enable to know the stop position for each discrete step time. While the different parameters are converging towards the right value, the final position assessment is becoming more and more precise. Here, the contact position is not accurate enough to enable to have very precise results; however, the results showed on the Annex E shows that this is working and very faster (after 1-2s). The prediction principle is based on a control on the covariance parameters and a trigger to execute the ODE with estimated parameters.

### 4.5 AUGMENTED REALITY

For the moment, the implemented algorithm with the Unscented Kalman Filter is fast enough to enable a real time augmented reality displayed on the result (a 0.9s execution for a 2s video with 50 frame/s).

In order to represent the friction coefficient and the contact points, the friction forces has been added on the ruler video. The representation of the final position is another example of the different hidden information that becomes visual with this method. See Annex F for an example.

### 4.6 DISCUSSIONS AND PERSPECTIVES

This preliminary work on the subject raises more questions than it solved. The concept works, and more quantitative results are quickly expected. For the moment, the robustness of the algorithm has to be tested deeply: the initial values used could be 50% false or more. In the same way, the repeatability on different event has to be tested on a larger batch of scenario: the initialization of the filter is one of the most difficult part of the process. The experimental part of the work with the image processing block is expected soon in order to make a comparative study.

## 5 CONCLUSION

Nowadays, there is a large scale development of object recognition based on SURF or SIFT algorithm ([23]) using a library containing the different signature of the object. This work could be linked with those developments, adding for each object a mechanical model describing the object behaviour, depending on its environment. The computer would recognize the object and know its potential dynamic behaviour.

## 6 ACKNOWLEDGEMENT

A special thanks to Vannier Kinoptik. The company has launched for 2 years the FAMOSYS project, FASt MOving SYStem, on which this work is related to. Vannier-Kinoptik is supporting this work by financing both with the ANRT (French National Technological Research Agency). Furthermore, Vannier-Kinoptik gives to the LISMMA laboratory all its cumulative experience in high speed camera (since 1966) in order to go as far as possible in this direction.

## 7 BIBLIOGRAPHY

- 1] G. Pilarski & C. Bouvier, «Soixante ans d'économie française : des mutations structurelles profondes,» *INSEE PREMIERE*, n°11201, July 2008.
- 2] G. Blackman, "Keeping European production competitive ?," *Imaging & machine vision Europe*, no. 62, pp. 20-23, April/May 2014.
- 3] C. E. Shannon, «Communication in the presence of noise,» *Proceedings of the Institute of Radio Engineers*, vol. 37, n°11, pp. 10-21, January 1949.
- 4] A. Yilmaz, O. Javed & M. Shah, «Object tracking: A survey,» *ACM Computing Surveys*, n°14, pp. 13-es, 2006-12-25.
- 5] Y. Benezeth, P.-M. Jodoin, B. Emile, H. Laurent & C. Rosenberg, «Comparative study of background subtraction algorithms,» *Journal of Electronic Imaging*, vol. 19, n°13, 2010-07-01.
- 6] S. J. Julier & J. K. Uhlmann, «Unscented Filtering and Nonlinear Estimation,» *Proceedings of the IEEE*, vol. 3, n°192, pp. 401-422, 03/2004.
- 7] S. Seibold & K. Weinert, «A Time Domain Method for the Localization of Cracks in Rotors.,» *Journal of Sound and Vibration*, n°1195, p. 57–73, 1996.
- 8] Z. Wang, W. Yan & Y. Shao, «Three-Step Method for Stiffness Identification of Inter-Story Shearing Structures Under Ambient Excitation.,» *Tsinghua Science & Technology*, n°114, pp. 69-74, 2009.
- 9] J.-M. Ma, C.-K. Lin & D.-C. Chang, «Estimation of forces generated by a machine mounted upon isolators under operating conditions.,» *Journal of the Franklin Institute*, n°1336, pp. 875-892, 1999.
- 10] C.-K. Ma & C.-C. Ho, «An inverse method for the estimation of input forces acting on non-linear structural systems,» *Journal of Sound and Vibration*, n°1275, p. 953–971, 2004.
- 11] Y. Zhan & A. Jardine, «Adaptive autoregressive modeling of non-stationary vibration signals under distinct gear states. Part 1: modeling,» *Journal of Sound and Vibration*, n°1286, pp. 429-450, 2005.
- 12] Y. Zhan & C. Mechefske, «Robust detection of gearbox deterioration using compromised autoregressive modeling and Kolmogorov–Smirnov test statistic—Part I: Compromised autoregressive modeling with the aid of hypothesis tests and simulation analysis,» *Mechanical Systems and Signal Processing*, n°121, pp. 1953-1982, 2007.
- 13] Y. Zhan & C. Mechefske, «Robust detection of gearbox deterioration using compromised autoregressive modeling and Kolmogorov–Smirnov test statistic. Part II: Experiment and application,» *Mechanical Systems and Signal Processing*, n°121, pp. 1983-2011, 2007.
- 14] Y. Shao & C. Mechefske, «Gearbox vibration monitoring using extended Kalman filters and hypothesis tests,» *Journal of Sound and Vibration*, n°1325, pp. 629-648, 2009.
- 15] G. Wang, Z. Luo, X. Qin, Y. Leng & T. Wang, «Fault identification and classification of rolling element bearing based on time-varying autoregressive spectrum,» *Mechanical Systems and Signal Processing*, n°122, pp. 934-947, 2008.
- 16] X.-J. Dong, G. Meng & J.-C. Peng, «Vibration control of piezoelectric smart structures based on system identification technique: Numerical simulation and experimental study,» *Journal of Sound and Vibration*, n°1297, pp. 680-693, 2006.
- 17] J. Marzbanrad, G. Ahmadi, H. Zohoor & Y. Hojjat, «Stochastic optimal preview control of a vehicle suspension,» *Journal of Sound and Vibration*, n°1275, pp. 973-990, 2004.
- 18] T. Yoshimura, K. Edokoro & N. Ananthanarayana, «An Active Suspension Model For Rail/Vehicle Systems With Preview and Stochastic Optimal Control,» *Journal of Sound and Vibration*, n°1166, pp. 507-519, 1993.
- 19] A. Farina, B. Ristic & D. Benvenuti, «Tracking a ballistic target: comparison of several nonlinear filters,» *Aerospace and Electronic Systems, IEEE Transactions on*, vol. 38, n°13, p. 854–867, 2002.
- 20] G. Siouris, G. Chen & J. Wang, «Tracking an incoming ballistic missile using an extended interval Kalman filter,» *IEEE Transactions on Aerospace and Electronic Systems*, vol. 33, n°11, pp. 232-240, 1997.
- 21] F. Naets, F. Cosco & W. Desmet, «An extended Kalman filter approach for augmented strain/stress visualization in mechanical system,» chez , 2014 *IEEE/ASME 10th International Conference on Mechatronic and Embedded Systems and Applications (MESA)*, 10-12 Sept. 2014.
- 22] R. E. Kalman, «A new approach to linear filtering and prediction Problems.,» *Transactions of the American Society of Mechanical Engineers Journal of Basic Engineering*, vol. Series D, n°182, pp. 35-45, 1960.
- 23] H. Bay, T. Tuytelaars & L. Van Gool, «Surf: Speeded up robust features,» *Computer Vision–ECCV 2006*, pp. 404-417, 2006.
- 24] I. T. Al-Zaharnah, «Suppressing vibrations of machining processes in both feed and radial directions using an optimal control strategy: The case of interrupted cutting,» *Journal of Materials Processing Technology*, n°1172, pp. 305-310, 2006.

## ANNEX A

### THE DIFFERENT SCENARIOS

Simulation Name	Version	Mu	L1	L2	m	g	L	x0	y0	Alpha0	Vx0	Vy0	VAlpha0
Ruler	3	0,2	0,1	0,1	0	10	1	0	0	0	3	1	10
Ruler	4	0,3	0,1	0,1	0	10	1	0	0	0	1	1	10
Ruler	5	0,3	0,4	0,4	0	10	1	0	0	0	2	3	1

Mu is the friction coefficient.

L1 and L2 are the distance of the contact point with the center.

m is the mass of the ruler.

g is the earth acceleration constant.

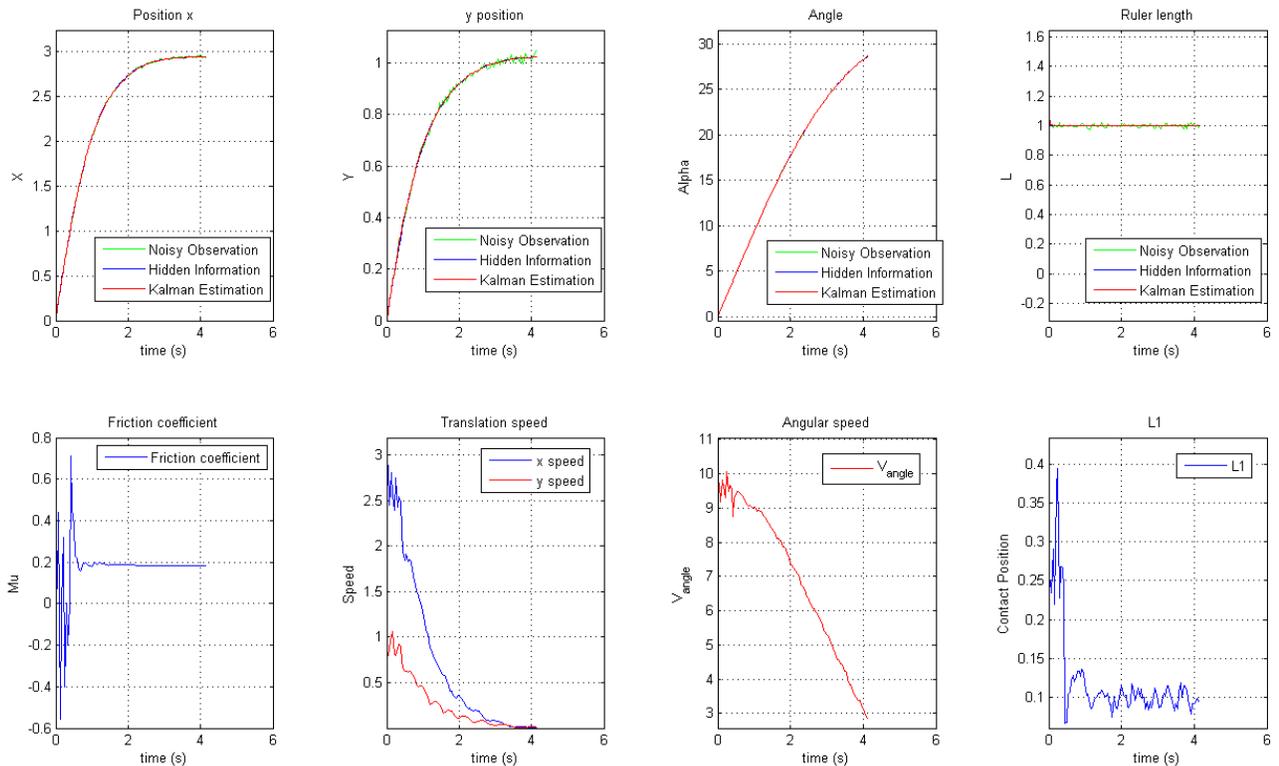
L is the ruler size.

x0, y0, Alpha0, Vx0, Vy0, VAlpha0 are the initial conditions simulated by the ruler shot simulator. These initial conditions are x0 and y0 for the position of the gravity center of the ruler, Alpha0 for the angle of the ruler, and Vx0, Vy0 and VAlpha0 the associated speed.

The units are not mentioned because the values chosen would not be really realistic. But L,L1, L2 could be meters as x0 y0, and Vx0 and Vy0 are  $m.s^{-1}$ , Alpha0 is radian and VAlpha0  $rad.s^{-1}$ , m is kilogram, and g is  $m.s^{-2}$ . Mu is a dry friction coefficient, so it has no dimension.

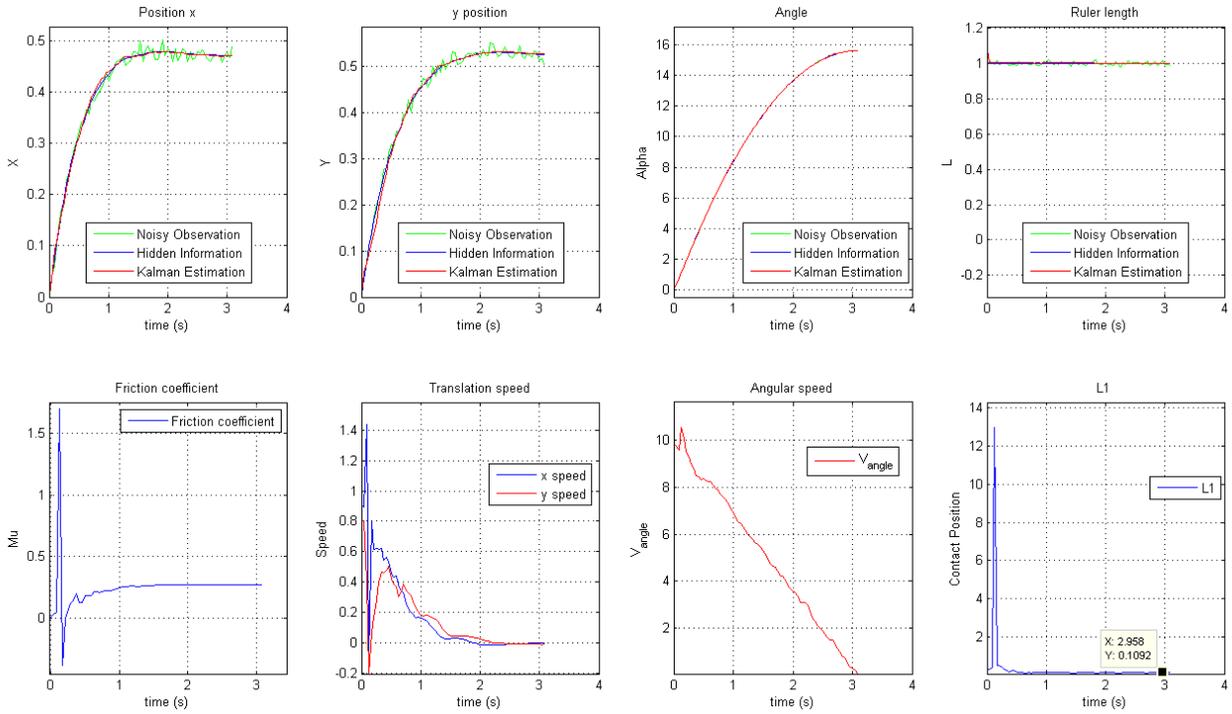
## ANNEX B

### DIAGRAMS FROM SCENARIO 3



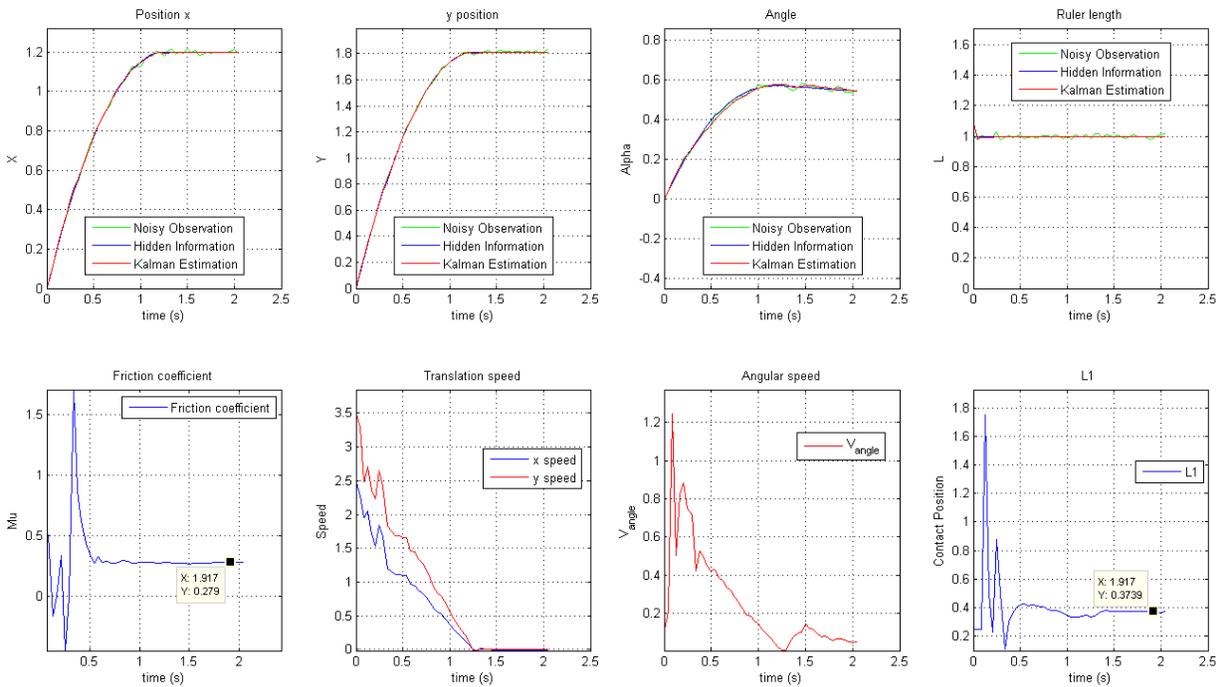
## ANNEX C

### DIAGRAMS FROM SCENARIO 4



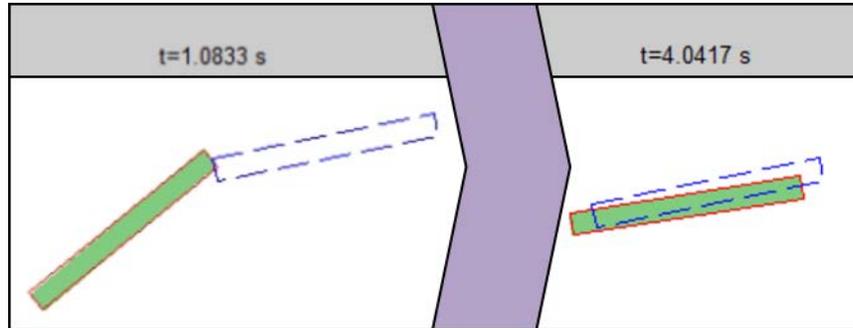
## ANNEX D

### DIAGRAMS FROM SCENARIO 5



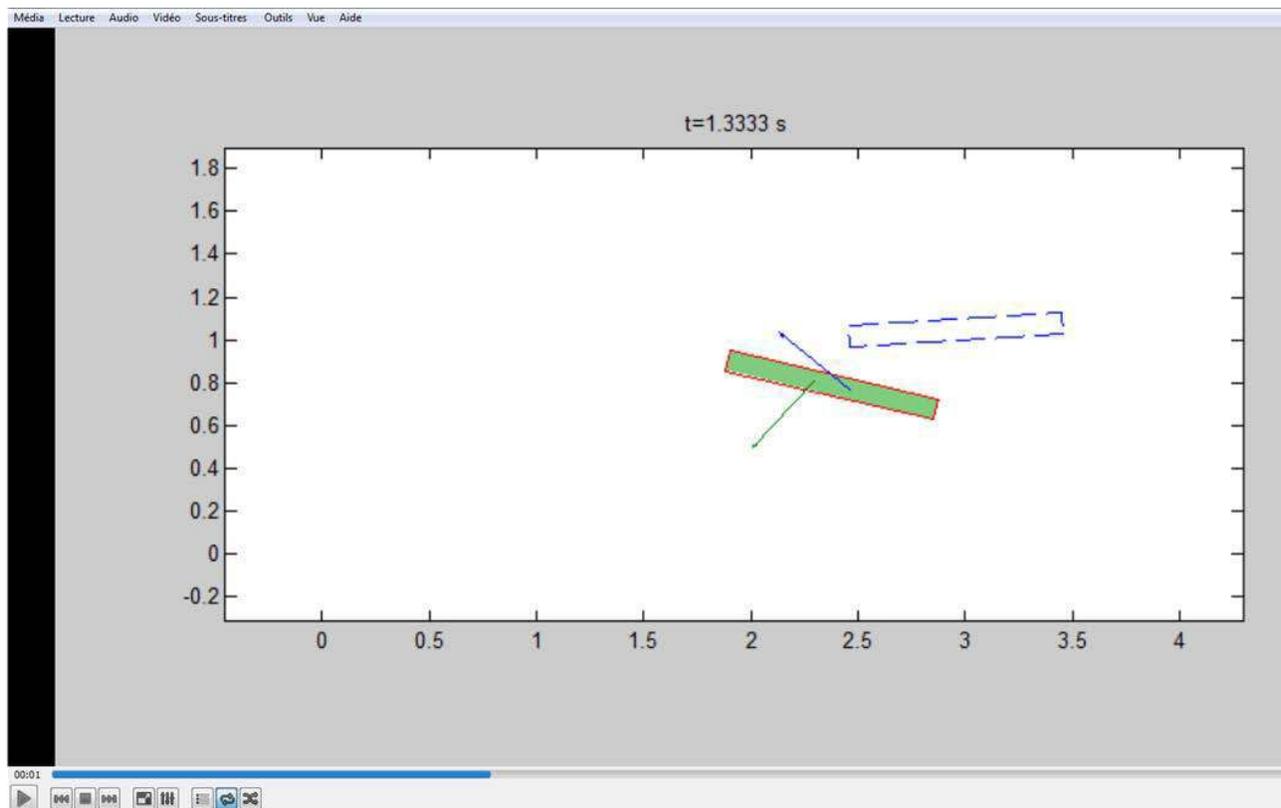
## ANNEX E

### PREDICTION EXAMPLE ON SCENARIO 3: VIDEO SNAPSHOTS AT THE TRIGGER OF THE FINAL POSITION AND ANOTHER AT THE FINALE POSITION



## ANNEX F

### AUGMENTED REALITY EXAMPLE: VIDEO SNAPSHOT FROM ESTIMATED SCENARIO 3



This annex shows the prediction from the Kalman filter in dashed blue, the simulated ruler in green, and the Estimated with the red edges. The two arrows are representing the friction forces on the contact points.