CENTER FOR
MACHINE PERCEPTION

CZECH TECHNICAL
UNIVERSITY IN PRAGUE

RESEARCH REPORT

ISSN 1213-2365

CTU Color and Depth Image Dataset of Spread Garments

Libor Wagner, Daniela Krejčová and
Vladimír Smutný

wagnelib@cmp.felk.cvut.cz

CTU–CMP–2013–25

October 3, 2013

The research leading to these results has recieved funding from the
European Communitys Seventh Framework Programme
(FP7/2007-2013) under grant agreement no. 288553

Research Reports of CMP, Czech Technical University in Prague, No. 25, 2013

Published by

Center for Machine Perception, Department of Cybernetics
Faculty of Electrical Engineering, Czech Technical University
Technická 2, 166 27 Prague 6, Czech Republic
fax +420 2 2435 7385, phone +420 2 2435 7637, www: http://cmp.felk.cvut.cz

CTU Color and Depth Image Dataset of Spread Garments

Libor Wagner, Daniela Krejčová and Vladimír Smutný

October 3, 2013

Abstract

This research report introduces a dataset consisting of colour and depth images of spread garments. The dataset is designed for testing and benchmarking of various computer vision algorithms. That might include cloth segmentation, garment recognition, model fitting and fold detection. Manually annotated ground truth is also provided.

1 Introduction

This datasets was created for the purposes of the Clothes Perception and Manipulation project (CloPeMa). CloPeMa is a 3 year open-source EU-FP7 research project which aims to advance the state of the art in the autonomous perception and manipulation of fabrics, textiles and garments. For that purpose a project goal was set to be an autonomous clothing processing. Starting from an unstructured laundry heap the output should be pile of folded garments.

The whole process of folding can be divided into number of steps. First, single garment is selected and picked up from the laundry heap by the robot. Second, the garment brought in some know configuration i.e. spread on a table. Third, the garment is folded info a rectangle.

This dataset is aimed at the algorithms involved in the second and third step of the garment processing. Color images can be used for segmentation, recognition and model fitting. The depth images can be used for example for wrinkle detection and spreading strategy estimation.

The dataset consists of RGB and Depth images of 17 different garments in various configurations. For the ground truth we have selected set of variables: type of the garment (the *class*), whether the garment is flat, wrinkled or folded (i.e. *state*), and whether the visible side is front or back (*facing*). For

each class we have defined a model as a list of corners, see Appendix for details.

We have used Assus Xtion PRO LIVE[1] placed on a tripod to capture color and depth images. As a background we have used a sheet of vinyl flooring. The overview of the setup can be seen in Figure 1.

Figure 1: The setup

2 Technical details

Each sample is identified by its number and is represented by three files; color image, depth map and annotation file.

Files are named according to the following template `cloXNNNNN.EXT` where `NNNNN` represents sample number and `X` can be one of `C`, `D` and `A` for color, depth and annotation respectively. The file extension `EXT` is `png` for color image and depth map, and `yaml` for annotation file.

Color images have resolution of 1280×1024 pixels, three channels and 8 bits per channel. Depth images have resolution of 640×480 pixels, one 16 bit channel[2].

[1] `http://www.asus.com/Multimedia/Xtion_PRO_LIVE/`

[2] Note that most of the image viewers are not capable of displaying 16 bit images.

The annotation uses YAML[3] format and the content can vary for different classes. The annotation contains the following fields.

path_c Path to color image relative to the annotation file.

path_d Path to depth image relative to the annotation file.

facing Denotes whether the garment is facing front or back.

shape Shape of the garment, whether it lies flat, wrinkled or folded.

type Defines the class of the garment i.e. *pants* or *towel*.

node_names Names of the corners in the model. Optionally it may contain folds. Fold is always made by two corners with the same name i.e. *fold_1*.

poly_c Positions of the corners in the color image, optionally it may contain fold corners position.

moves Optional description of simple movements that should bring the wrinkled garment to the flat state.

3 Tools

In order to simplify the start with this dataset we have prepared a simple visualisation tool in Matlab. The source code and source for this document is available on the GitHub (`https://github.com/CloPeMa/garment_dataset`).

4 Notes

The second part of dataset, the folded garments, was annotated using different tool that allow for more precise localisation of the control points.

[3] `http://yaml.org`

A Classes and models

We have divided garments into nine classes based on their visual similarities. The classes are *blouse*, *hoody*, *pants*, *polo*, *polo-long*, *skirt*, *towel*, *tshirt*, *tshirt-long*. For each class we have defined a model as a list of named points. The points forms a polygon that roughly corresponds to the garment outline.

We have divided short and long sleeved garments into different classes even though their model is identical. This way the model fitting algorithm can use the sleeve length as a clue or treat the two classes as one.

A.1 Blouse

The *blouse* class represents tops with distinct collar, without sleeves and without straight bottom line.

#	name
1	bottom-left
2	bottom-middle
3	bottom-right
4	right-armpit
5	right-shoulder
6	neckline-right
7	collar-right
8	collar-left
9	neckline-left
10	left-shoulder
11	left-armpit

Table 1: Blouse corner names

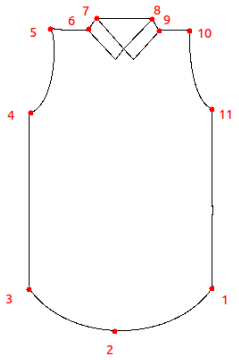

Figure 2: Blouse model

Figure 3: *blouse*, the only instance of *blouse* class

A.2 Hoody

The *hoody* class represents hooded tops with long sleeves.

#	name
1	bottom-left
2	bottom-right
3	right-armpit
4	right-sleeve-inner
5	right-sleeve-outer
6	right-shoulder
7	hood-right
8	hood-top
9	hood-left
10	left-shoulder
11	left-sleeve-outer
12	left-sleeve-inner
13	left-armpit

Table 2: Pants mode definition

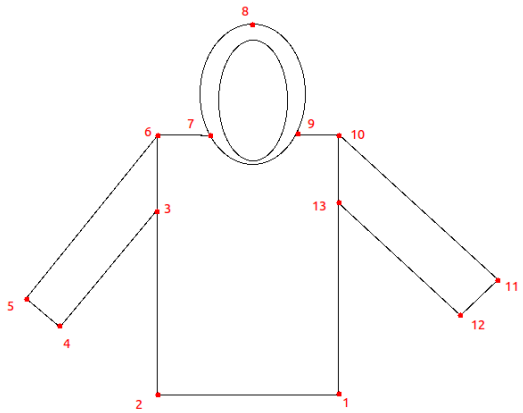

Figure 4: The setup

Figure 5: *hoody*, the only instance of *hoody* class

A.3 Pants

The *pants* class can cover almost all bottoms with long legs.

#	name
1	left-leg-outer
2	left-leg-inner
3	crotch
4	right-leg-inner
5	right-leg-inner
6	top-right
7	top-left

Table 3: Pants mode definition

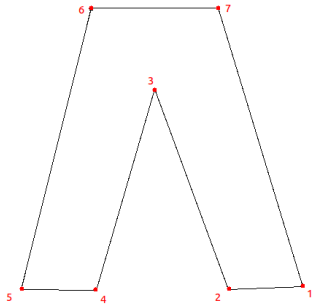

Figure 6: The setup

(a) *jeans_blue*

(b) *jeans_darkblue*

(c) *trausers_brown*

(d) *trausers_gray*

Figure 7: Instances of the *pants*

A.4 Polo

Under *polo* class fits every top with distinct collar and with short sleeves.

#	name
1	bottom-left
2	bottom-right
3	right-armpit
4	right-sleeve-inner
5	right-sleeve-outer
6	right-shoulder
7	neckline-right
8	collar-right
9	collar-left
10	neckline-left
11	left-shoulder
12	left-sleeve-outer
13	left-sleeve-inner
14	left-armpit

Table 4: Pants mode definition

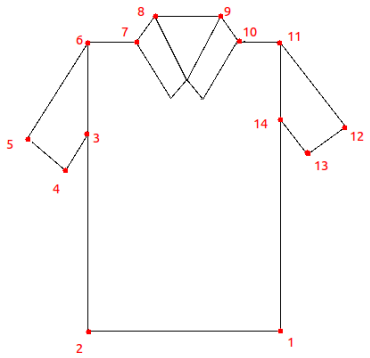

Figure 8: The *pants* model

(a) *shirt_chequered* **(b)** *poloshirt_gray*

Figure 9: Instances of the *polo* class

A.5 Long sleeved polo

The *polo_long* class is similar to the normal *polo* but with long sleeves.

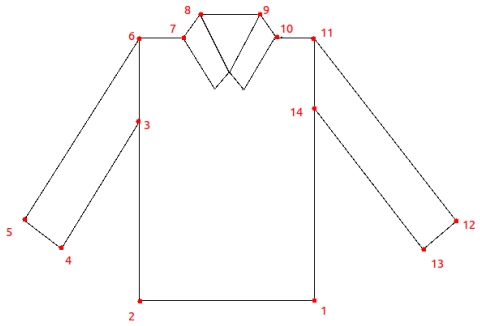

Figure 10: The *polo_long* model

(a) *shirt_flowers* **(b)** *coat_knitted*

Figure 11: Instances of the *polo_long* class

A.6 Skirt

The *skirt* class.

#	name
1	bottom-left
2	bottom-right
3	top-right
4	top-left

Table 5: Skirt mode definition

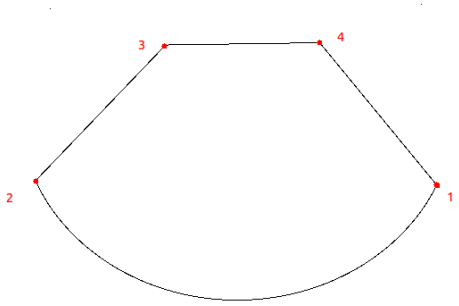

Figure 12: Skirt model

Figure 13: *skirt*, the only instance of *skirt* class

A.7 Towel

The *towel* class encapsulates every rectangular piece cloth.

#	name
1	bottom-left
2	bottom-right
3	top-right
4	top-left

Table 6: Towel mode definition

Figure 14: Towel visual model

Figure 15: *towel*, the only instance of *towel* class

A.8 T-shirt

The *tshirt* class describes a top without distinct collar with short sleeves.

#	name
1	bottom-left
2	bottom-right
3	right-armpit
4	right-sleeve-inner
5	right-sleeve-outer
6	right-shoulder
7	neckline-right
8	neckline-left
9	left-shoulder
10	left-sleeve-outer
11	left-sleeve-inner
12	left-armpit

Table 7: T-shirt corner names

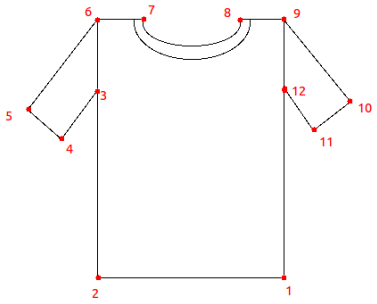

Figure 16: T-shirt model

(a) *tshirt_yellow* **(b)** *jumper_shortsleeved*

Figure 17: Instances of the *tshirt* class

A.9 Long Sleeved T-shirt

The *tshirt_long* class is similar to the *tshirt* but with long sleeves.

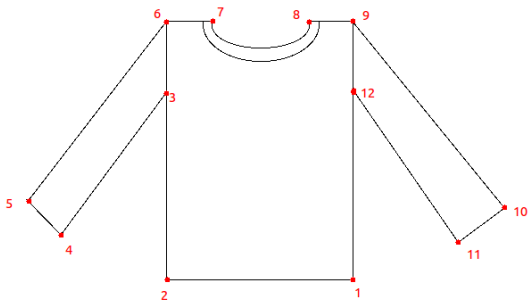

Figure 18: Long sleeved t-shirt model

(a) *sweater_violet* **(b)** *jumper_green*

Figure 19: Instances of the *tshirt-long* class