



# DeepSight Toolkit Feature Documentation

v6.2.1





### **TABLE OF CONTENTS**

#### 1. INTRODUCTION

#### 2. Using the CSV DUMPER

2.1. Item description - Raw data CSV

2.2 Aggregated CSV - Aggregated data CSV

#### 3. Using the PULL MECHANISM

3.1. JSON raw example output

3.2 JSON aggregated example output

#### 4. Using the PUSH MECHANISM

4.1. JSON raw example output

4.2 JSON aggregated example output

5. Using the AUTO-START

#### 6. Automatic OS Updates

6.1. Schedule downloads and OS updates

6.2. Disable OS updates





#### **1. INTRODUCTION**

Welcome to the DeepSight Toolkit feature documentation! This guide contains technical descriptions of the newest Toolkit features and step-by-step instructions on how to use them.

The following sections explain how to integrate BI or CMS platforms directly with the software and how to auto-start the application after system reboot.

DeepSight Toolkit empowers you with accurate, real-time insight into your audience as they interact with your brand in real-world environments. It enables you to discover their behavior patterns, interests, and demographic profiles. With this actionable data at your fingertips, you are free to focus on the things you love: engaging your customers, optimizing your business, and making data-driven decisions.

#### **Key Features:**

- ✓ Easy & quick setup
- ✓ Affordable pricing
- ✓ CMS Integration
- ✓ Real-time results
- $\checkmark$  Offline analysis

- $\checkmark$  Anonymous analysis
- ✓ Cross-Platform
- ✓ Processed Locally
- ✓ Push mechanism
- $\checkmark$  Compatible with most cameras





### 2. Using the CSV DUMPER

A CSV file is a *comma separated values* file which allows data to be saved in a table structured format. The Toolkit generates two CSV files: one containing **raw data** gathered during the real-time analysis, and another with **pre-aggregated data** for each person ID. The raw data can be used for post-processing.

🕑 💿 🎤 💾	FPS
General Settings Input Input Settings Output Licensing About	
Location Label you_location_name	
Raw data	
Enable Pull Access JSON at http://localhost: 8080 /JSON	
Enable Push http://localhost:8081      Compatibility mode [Deprecated]	1 🚺
Save to CSV C:/CSV_raw	
Aggregated data	
Enable Pull Access JSON at http://localhost: 8080 /JSONAGGR Frequency 30 sec 💌	
Enable Push http://localhost:8081	
Save to CSV C:/CSV_aggregated	

To enable CSV exports go to the *Output* tab and make sure the checkbox **Save data to CSV** is checked. By default, the CSV files will be automatically generated in the Toolkit configuration folder after each analysis. Users can also specify the location where the CSV files will be stored. Every row in the **raw** data CSV file contains the analysis for **1 frame**. In the **aggregated** CSV file each row contains data for one **individual ID**/person.



Below is an example CSV file (raw). The first row of the CSV file contains the label of the respective column.

	Α	В	С	D	E	F	G	н	1	J		к	L	М	N	0	Р	Q	R	S	т	U	V	W
1	frame	timestamp	id a	ge ge	nder p	oitch	yaw	roll	is_looking	gaze	x gaz	ze_y	detection_time	detection_duration	total_attention_duration	current_attention_duration	smile	bounding_box_x	bounding_box_y	bounding_box_width	bounding_box_height	world_x	world_y	world_z
2	1	1570610706	12	46	-95	-6	-17	-17	1	21	.6	-63	1.57061E+16	1362512	1362512	1362512	3	3 7	238	97	127	144	38	242
3	2	1570610706	12	46	-89	-5	-24	-18	1	23	9	-58	1.57061E+16	1362668	1362668	1362668	4	1 18	240	99	121	139	37	237
4	3	1570610706	12	45	-90	-3	-12	-18	1	19	2	-51	1.57061E+16	1362830	1362830	1362830	3	3 26	239	97	119	141	38	242
5	4	1570610706	12	45	-90	-5	-16	-19	1	21	2	-60	1.57061E+16	1362989	1362989	1362989	4	1 33	239	95	123	143	39	247
6	5	1570610706	12	44	-91	-8	-25	-18	0	24	14	-71	1.57061E+16	1363137	1362989	0	4	42	241	97	121	139	38	242
7	6	1570610706	12	44	-91	-6	-20	-17	1	22	29	-65	1.57061E+16	1363312	1363164	175	4	1 51	. 241	94	120	142	39	249
8	7	1570610706	12	44	-91	-5	-19	-18	1	21	7	-59	1.57061E+16	1363459	1363311	322	4	1 66	241	96	120	136	38	244
9	8	1570610707	12	44	-91	-3	-14	-17	1	19	8	-52	1.57061E+16	1363610	1363462	473	4	1 77	242	94	120	137	39	249
10	9	1570610707	12	44	-91	-6	-18	-17	1	22	20	-66	1.57061E+16	1363791	1363643	654	4	1 84	243	92	122	139	39	255
11	10	1570610707	12	43	-91	-5	-21	-18	1	22	28	-61	1.57061E+16	1363938	1363790	801	5	5 101	244	93	123	135	39	252
12	11	1570610707	12	43	-91	-4	-17	-16	1	20	14	-55	1.57061E+16	1364082	1363934	945	4	108	245	95	122	131	38	247
13	12	1570610707	12	44	-91	-4	-19	-12	1	22	21	-57	1.57061E+16	1364224	1364076	1087	4	1 117	248	91	118	135	39	258
14	13	1570610707	12	43	-91	-5	-33	-17	0	27	70	-58	1.57061E+16	1364377	1364076	0	4	1 127	251	95	123	128	37	247
15	14	1570610708	12	43	-91	-4	-19	-14	1	21	3	-55	1.57061E+16	1364552	1364251	175	4	1 136	249	93	125	129	38	252
16	15	1570610708	12	43	-91	-8	-27	-15	0	25	<b>0</b>	-73	1.57061E+16	1364697	1364251	0	5	5 144	252	92	124	129	38	255
17	16	1570610708	12	43	-91	-7	-24	-16	1	22	27	-66	1.57061E+16	1364858	1364412	161	5	5 151	255	95	121	124	36	247
18	17	1570610708	12	43	-91	-9	-24	-13	1	23	9	-79	1.57061E+16	1364999	1364553	302	6	5 160	255	90	123	130	38	260
19	18	1570610708	12	43	-91	-4	-25	-9	0	23	3	-54	1.57061E+16	1365171	1364553	0	4	1 177	255	93	125	122	37	252
20	19	1570610708	12	44	-92	-7	-23	-10	1	22	25	-68	1.57061E+16	1365316	1364698	145	4	1 184	256	92	126	122	37	255
21	20	1570610708	12	44	-92	-7	-25	-10	0	23	5	-69	1.57061E+16	1365471	1364698	0	4	4 193	256	91	126	122	37	258
22	21	1570610709	12	44	-92	-6	-34	-10	0	2	0	-63	1.57061E+16	1365629	1364698	0	5	5 208	259	92	121	118	37	255
23	22	1570610709	12	44	-92	-4	-26	-10	0	23	0	-54	1.57061E+16	1365781	1364698	0	5	5 217	255	93	124	116	37	252
24	23	1570610709	12	43	-92	-4	-27	-10	0	23	1	-54	1.57061E+16	1365946	1364698	0	4	1 226	254	94	123	113	37	249

Example raw data CSV Export

**CSV tip**: Instead of double clicking on the CSV file to open it with Excel, we recommend opening a blank Excel file and clicking on 'Data' -> 'From Text/CSV'. This will import the data separated in columns as shown in the example, instead of a single column with values separated by commas.





The raw data CSV example shown above can be used for analysis, however, if you are not familiar with Python or R, you can use the aggregated CSV which can be easily filtered in Excel.

	Α	В	С	D	E	F	G	Н	I	J	К	L	М
1	id	timestamp	age	gender	impressions	views	attention_sessions	average_attention_time	attention_time	dwell_time	total_detection_time	smile_viewing	smile_overall
2	1	2019/12/17 9/40/56	24	f	1	1	3	0.305	0.916	8.839	8.839	5	5
3	2	2019/12/17 9/40/56	33	f	1	1	2	2.371	4.742	8.276	8.276	32	32
4	3	2019/12/17 9/40/56	-1	u	1	1	1	0.159	0.159	6.943	6.943	7	7
5	4	2019/12/17 9/40/57	38	m	1	0	0	0	0	7.492	7.492	0	24
6	5	2019/12/17 9/40/59	-1	u	1	1	3	0.362	1.087	2.2	2.2	2	7
7	7	2019/12/17 9/41/1	-1	u	1	1	1	0.135	0.135	1.222	1.222	6	15
8	8	2019/12/17 9/41/3	-1	u	1	0	0	0	0	1.16	1.16	0	29
9	10	2019/12/17 9/41/5	7	m	1	0	0	0	0	4.474	4.474	0	15
10	11	2019/12/17 9/41/6	45	f	1	0	0	0	0	3.613	3.613	0	16
11	12	2019/12/17 9/41/6	-1	u	1	0	0	0	0	2.616	2.616	0	25
12	13	2019/12/17 9/41/6	-1	u	1	1	10	0.412	4.121	10.978	10.978	20	20
13	14	2019/12/17 9/41/7	46	f	1	1	4	0.846	3.384	6.944	8.686	26	26
14	15	2019/12/17 9/41/8	32	m	1	0	0	0	0	3.332	3.332	0	21
15	16	2019/12/17 9/41/9	40	m	1	1	4	0.212	0.85	2.663	2.663	6	44
16	19	2019/12/17 9/41/13	20	m	1	1	5	0.649	3.249	22.712	22.712	5	5
17	21	2019/12/17 9/41/14	41	f	1	1	2	0.61	1.22	1.516	1.516	5	5
18	23	2019/12/17 9/41/16	13	m	1	1	13	0.551	7.164	32.529	32.529	5	29
19	25	2019/12/17 9/41/18	35	f	1	1	5	0.201	1.008	4.245	4.565	7	21

Example aggregated data CSV Export

As shown above, the aggregated CSV output is easier to read and already contains some average and total values grouped by ID. This takes the difficult part out of the data analysis, allowing you to simply filter columns in Excel to get actionable data.



To know more about the Toolkit CSV output you can download a sample file below:



Every row in the raw data CSV file contains the analysis for **each frame**. In the aggregated CSV file each row contains data for **each individual ID/person**.

You can also watch a short walk-though video on how to use the aggregated CSV file.





#### 2.1. Item description - Raw data CSV

**Frame**: this is the frame number as analysed from the video feed. The analysed video is divided in single frames, and each frame is given a number in the sequence. Therefore, the first frame being analysed is frame 1, the second is frame 2 and so on. **Only frames with faces analyzed are saved to the CSV.** If there is no face detected in a frame, it will be skipped in the CSV.

**Timestamp**: this is the **time** when the frame was analysed, according to the PC clock. The time is expressed in **Unix timestamp** as it is the most widely accepted way for expressing time. There are services like **this** to convert from Unix timestamp to a human readable format. Excel, other tools, and programming languages, can also understand and convert the timestamp format to other formats. One frame is relative to one moment in time and therefore it appears in only one timestamp, but in one timestamp multiple frames can be analysed. **Multiple frames can be analysed in the same second**. If the frame rate (FPS) is 10, then there will be 10 frames in the same timestamp (timestamp is expressed in seconds).

**ID**: this is the unique identification number assigned by the Toolkit to a **detected person**. This ID number is only temporary therefore if someone exits the frame and then returns after some time, they will be assigned a new ID. This way the analysis remains anonymous and there is no re-identification of people.



Age: This value represents the age estimation per detection and ranges from 0 to 99 in years.

**Gender**: The confidence of a detection belonging to a certain gender category. It ranges from **-99 to 99**, where **-99** is considered high confidence for **male**, and **99** high confidence for **female**. Estimations closer to 0 are low confidence / unknown.

**Headpose**: The direction of where a person is looking is represented by head pose estimation using **pitch**, **yaw**, **and roll** in degrees (see below).







**Is\_looking**: A binary value (1 or 0) that records if a person is looking or not looking towards the point of interest (camera). When is\_looking=1, the values for total\_attention and current\_attention will start counting the duration in milliseconds. This value can be used to distinguish between viewers and passersby.

**Total\_attention**: Cumulative value for which an ID was detected as is\_looking=1 expressed in milliseconds.

**Current\_attention**: The duration of current attention session expressed in milliseconds. This value is only logged when is\_looking=1. When is\_looking=0, this value will reset back to 0. This allows users to separate the total attention time into individual sessions.

**Detection\_time**: Time stamp at which the face detection occurred. This value is unique to each detected face. Unlike the first 'timestamp' column, there cannot be multiple faces analyzed within the same 'detection\_time' entry.

**Detection\_duration**: The total cumulative time during which an ID was detected expressed in milliseconds. This value can be used to measure Opportunity to see (OTS).





**Mood/smile**: This value represents the amount of smiling and ranges from 0 to 100.

**Gaze\_x, Gaze\_y**: The position of a person's gaze with regards to the point of interest (camera). The value is expressed as a single point.

**Bounding\_box\_x**: top left x coordinate in pixels of the face location.

**Bounding\_box\_y**: top left y coordinate in pixels of the face location.

**Bounding\_box\_width**: the width of the rectangle enclosing the detected face.

**Bounding\_box\_height**: the height of the rectangle enclosing the detected face.



### 2.2. Item description - Aggregated data CSV

**ID**: this is the unique identification number assigned by the Toolkit to a **detected person**. This ID number is only temporary therefore if someone exits the frame and then returns after some time, they will be assigned a new ID. This way the analysis remains anonymous and there is no re-identification of people.

**Timestamp**: the date and time at which an ID was first detected. This is expressed in a dd/mm/yyyy hh:mm:ss format.

**Age**: the age value for a specific ID.

**Gender**: the gender value for a specific ID (m = male, f = female, u = unknown).

**Impressions**: the number of people detected as a qualified impression - head pose range within the  $+45^{\circ}$  /  $-45^{\circ}$ .

**Views**: a binary value that changes to 1 whenever a person's head position is detected as looking towards the focal point (camera). The viewing angle changes with the distance from the camera. Detections further away from the camera will have a narrower 'Viewing' angle which provides far more accurate estimations.



Attention sessions: the number of times a detected person has been 'Viewing'. Dividing the number of attention sessions by the number of viewers gives you an average number of views per ID.

**Average attention time**: the average duration of an attention sessions per ID.

Attention time: the total amount of time that a detected person has spent looking towards the object of interest (camera). This is expressed in seconds with 3 decimals.

**Dwell time**: the total amount of time that a detected person has spent facing the object of interest (camera). A person is considered as facing the camera whenever their head pose (yaw) is between +45° and -45°.

**Total Detection time**: cumulative detection duration per ID expressed in seconds with 3 decimals.

**Smile viewing**: the maximum smile value while a person was 'Viewing' expressed in %.

**Smile maximum**: the maximum smile value (in %) while a person was detected. This can be compared to the maximum 'Smile when viewing' to asses the effect of the displayed content on the overall mood of your audience.



### **3. Using the PULL MECHANISM**

The HTTP interface can be used to connect any Business Intelligence platform or CMS to the Toolkit. Go to '**Output**' Tab and click on the fields: '**Enable Pull**'. You have the option to either get **raw** or **aggregated** data.

The JSON response can be accessed through a GET request on localhost for your external application. You can use this function for instance for Digital Signage or narrowcasting applications which trigger custom and relevant ad content to the right people at the right time.

For the raw data JSON output you can access: http://localhost:8080/JSON For the aggregated JSON output you can access: http://localhost:8080/JSONAGGR

On the next page you can find a screenshot highlighting the Output tab along with the boxes that need to be checked for the HTTP interface to function.





	FPS
General Settings Input Input Settings Output Licensing About	
Location Label       Enter name to uniquely identify this location         Raw data	
Enable Pull Access JSON at http://localhost: 8085 /JSON	~
Enable Push http://your.dashboard.endpoint.raw     Compatibility mode [Deprecated]	0
Save to CSV C:/Users/sightcorp/Desktop	
Aggregated data	
Enable Pull Access JSON at http://localhost: 8085 /JSONAGGR Frequency 30 sec 💌	
Enable Push http://your.dashboard.endpoint.aggregated	
Save to CSV C:/Users/sightcorp/Desktop	

Toolkit Output tab



### 3.1. JSON raw example output

{	{
"source": <camera_name>,</camera_name>	"source": "cam_01",
"timestamp": <unix in="" milliseconds≻,<="" td="" timestamp=""><td>"timestamp": 1570456789328,</td></unix>	"timestamp": 1570456789328,
"frame_number": <frame 0="" from="" in="" number="" sequence="" starting=""/> ,	"frame_number": 0,
"people": [	"people": [
{	{
"id": <person id="">,</person>	"id": 1,
"age": <age estimation="">,</age>	"age": 28,
<pre>"gender": <gender 100="female}" estimation="" {-100="male,">,</gender></pre>	"gender": -98,
"smile": <quantity 100="100%" of="" smile,="" smiling}="" {0="neutral,">,</quantity>	"smile": 27,
<pre>"bounding_box": {</pre>	<pre>"bounding_box": {</pre>
"x": ≺x coordinate in pixels>,	"x": 251,
"y": ≺y coordinate in pixels>,	"y": 246,
"w": ≺width in pixels>,	"w": 155,
"h": <height in="" pixels=""></height>	"h": 229
},	},
"headpose": {	"headpose": {
"yaw": ≺yaw angle in degrees>,	"yaw": 1,
"pitch": <pitch angle="" degrees="" in="">,</pitch>	"pitch": -2,
"roll": <roll angle="" degrees="" in=""></roll>	"roll": 0
},	},
<pre>"is_looking": &lt;{0 = not looking, 1 = looking}&gt;,</pre>	"is_looking": 1,
"detection_duration": <total a="" being="" detected<="" duration="" of="" person="" td=""><td>"detection_duration": 28652,</td></total>	"detection_duration": 28652,
in milliseconds>,	"current_attention_duration": 28652,
"current_attention_duration": <current a="" duration="" is<="" person="" td="" uninterrupted=""><td>"total_attention_duration": 28652,</td></current>	"total_attention_duration": 28652,
looking at stimulus in milliseconds>,	"gaze": {
"total_attention_duration": <total a="" at<="" duration="" is="" looking="" person="" td=""><td>"x": -8,</td></total>	"x": -8,
stimulus in milliseconds≻,	"y": 10
"gaze": {	}
"x": -8,	}
"y": 10	]
}	}
}	
}	



### 3.2. JSON aggregated example output

"submission_time": <date_time dd="" hh:mm:ss="" in="" mm="" yyyy="">,</date_time>	"submission_time": "2019/12/17 15:56:49",
"detection_count": <number detected="" of="" people="">,</number>	"detection_count": 1,
"source": <location_label>,</location_label>	"source":"Camera 01",
<pre>"aggregator_freq": <frequency at="" data="" in="" is="" milliseconds="" reported="" which="">,</frequency></pre>	"aggregator_freq": 60000,
"aggregated_data": [{	"aggregated_data": [{
"id": <person id="">,</person>	"id": 1,
"timestamp": <date_time dd="" hh:mm:ss="" in="" mm="" yyyy="">,</date_time>	"timestamp": "2019/12/17 15:55:49",
"age": <age estimation="">,</age>	"age": 34,
<pre>"gender": <gender estimation="" f="female," u="unknown}" {m="male,">,</gender></pre>	"gender": "m",
"impressions": <count impressions="" of="">,</count>	"impressions": 2,
"views": <count of="" views="">,</count>	"views": 1,
"attention_sessions": <count attention="" of="" sessions="">,</count>	"attention_sessions": 1,
"average_attention_time": <average attention="" in="" miliseconds="" per="" session="" time="">,</average>	"average_attention_time": 1.300,
"attention_time": <total attention="" in="" milliseconds="" time="">,</total>	"attention_time": 1.300,
"dwell_time": <total dwelling="" in="" milliseconds="" time="">,</total>	"dwell_time": 5.451,
"total_detection_time": <total detection="" in="" milliseconds="" time="">,</total>	"total_detection_time": 5.646,
"smile_viewing": <maximum smile="" viewing="" when="">,</maximum>	"smile_viewing": 21,
"smile_maximum": <maximum overall="" smile=""></maximum>	"smile_maximum": 34
}]	}]
}	}



### 4. Using the PUSH Mechanism

The push mechanism allows for messages to be sent from the Toolkit to external applications (eg. **CMS systems**) instead of requiring the external application to actively ask the Toolkit for changing status (pulling). Therefore the Toolkit can push data into any external application or API.

If the push feature is enabled, the Toolkit will send an **HTTP POST** request to the endpoint defined by the user every second. The endpoint needs to be a **valid HTTP server** able to receive HTTP POST requests. The body of the HTTP POST request is a JSON object containing a summary of the last second of analysis. You can add the target URL directly in the Toolkit interface.

<ul> <li>●</li> <li>●</li> <li></li> </ul>	FPS
General Settings   Input   Input Settings   Output   Licensing   About	
Location Label you_location_name	
Raw data	
Enable Pull Access JSON at http://localhost: 8080 /JSON     Send image data     Compatibility mode [Deprecated]	0
Enable Push http://localhost:8081	
Save to CSV C:/CSV_raw	
Aggregated data	
Enable Pull Access JSON at http://localhost: 8080 /JSONAGGR Frequency 30 sec 💌	
Enable Push http://localhost:8081	
Save to CSV C:/CSV_aggregated	

Toolkit Output tab





The aggregated JSON output will be sent to your server's location every 30 seconds, 1 minute, 5 minutes, or 10 minutes, depending on the **Frequency** that you select from the drop down box on the right hand side of the *Output* tab. This will contain data aggregated per ID for any detections that occurred during the aggregation period.

The Toolkit will aggregate data for each ID 5 seconds after the last detection occurs in case the same person returns into the video frame.

🖲 💽 🎦 🗎	FPS
General Settings   Input   Input Settings   Output   Licensing   About	
Location Label you_location_name	
Raw data	
Enable Pull Access JSON at http://localhost: 8080 /JSON Send image data	
Enable Push http://localhost:8081	0
Save to CSV C:/CSV_raw	
Aggregated data	
Enable Pull Access JSON at http://localhost: 8080 /JSONAGGR Frequency 30 sec 💌	
Enable Push http://localhost:8081	
Save to CSV C:/CSV_aggregated	

#### Toolkit Output tab



#### 4.1. JSON raw example output

```
"source": <camera name>,
                                                                                  "source": "cam_01",
"timestamp": <unix timestamp in milliseconds>,
                                                                                  "timestamp": 1570456789328,
"frame_number": <frame number in sequence starting from 0>,
                                                                                  "frame_number": 0,
"people": [
                                                                                  "people": [
    "id": <person ID>,
                                                                                      "id": 1,
   "age": <age estimation>,
                                                                                      "age": 28,
    "gender": <gender estimation {-100 = male, 100 = female}>,
                                                                                      "gender": -98,
    "smile": <quantity of smile, {0 = neutral, 100 = 100% smiling}>,
                                                                                      "smile": 27,
    "bounding_box": {
                                                                                      "bounding_box": {
     "x": <x coordinate in pixels>,
                                                                                        "x": 251,
     "y": <y coordinate in pixels>,
                                                                                        "y": 246,
      "w": <width in pixels>,
                                                                                        "w": 155,
      "h": <height in pixels>
                                                                                        "h": 229
    "headpose": {
                                                                                      "headpose": {
                                                                                        "yaw": 1,
      "yaw": <yaw angle in degrees>,
                                                                                        "pitch": -2,
      "pitch": <pitch angle in degrees>,
      "roll": <roll angle in degrees>
                                                                                        "roll": 0
    "is looking": <{0 = not looking, 1 = looking}>,
                                                                                      "is_looking": 1,
    "detection_duration": <total duration of a person being detected
                                                                                      "detection_duration": 28652,
        in milliseconds>,
                                                                                      "current attention duration": 28652,
    "current_attention_duration": <current uninterrupted duration a person is
                                                                                      "total_attention_duration": 28652,
       looking at stimulus in milliseconds>,
                                                                                      "gaze": {
    "total attention duration": <total duration a person is looking at
                                                                                        "x": -8,
        stimulus in milliseconds>,
                                                                                        "y": 10
    "gaze": {
      "x": -8,
      "y": 10
```



### 4.2. JSON aggregated example output

{	{
"submission_time": <date_time dd="" hh:mm:ss="" in="" mm="" yyyy="">,</date_time>	"submission_time": "2019/12/17 15:56:49",
"detection_count": <number detected="" of="" people="">,</number>	"detection_count": 1,
"source": <location_label>,</location_label>	"source":"Camera 01",
<pre>"aggregator_freq": <frequency at="" data="" in="" is="" milliseconds="" reported="" which="">,</frequency></pre>	"aggregator_freq": 60000,
"aggregated_data": [{	"aggregated_data": [{
"id": <person id="">,</person>	"id": 1,
"timestamp": <date_time dd="" hh:mm:ss="" in="" mm="" yyyy="">,</date_time>	"timestamp": "2019/12/17 15:55:49",
"age": <age estimation="">,</age>	"age": 34,
<pre>"gender": <gender estimation="" f="female," u="unknown}" {m="male,">,</gender></pre>	"gender": "m",
<pre>"impressions": <count impressions="" of="">,</count></pre>	"impressions": 2,
"views": <count of="" views="">,</count>	"views": 1,
"attention_sessions": <count attention="" of="" sessions="">,</count>	"attention_sessions": 1,
"average_attention_time": <average attention="" in="" miliseconds="" per="" session="" time="">,</average>	"average_attention_time": 1.300,
"attention_time": <total attention="" in="" milliseconds="" time="">,</total>	"attention_time": 1.300,
"dwell_time": <total dwelling="" in="" milliseconds="" time="">,</total>	"dwell_time": 5.451,
"total_detection_time": <total detection="" in="" milliseconds="" time="">,</total>	"total_detection_time": 5.646,
"smile_viewing": <maximum smile="" viewing="" when="">,</maximum>	"smile_viewing": 21,
"smile_maximum": <maximum overall="" smile=""></maximum>	"smile_maximum": 34
}]	}]
}	}





### 5. Using the AUTO-START

It might be desirable to start the Toolkit on startup of the system. For both platforms it is required that you log in automatically when the system is turned on, eg. you are not required to enter a password and you're instantly booted to the Desktop.

#### Windows 10

In the installation directory of the Toolkit there is a default **autorun.bat** script that can be used and modified for automatic startup of the Toolkit from the command line.

Once the **autorun.bat** script has been modified in accordance to your preference, you can use this script to automatically start up the Toolkit as follows:

- 1. Use the Windows key + R keyboard shortcut to open the Run command;
- 2. Type shell:startup to open the Startup folder of Windows. All shortcuts placed in this directory will be called when Windows has started;
- 3. Right-click and select New > Shortcut;
- 4. Browse and select the autorun.bat file;
- 5. Double-check if the Toolkit starts up when you double-click the shortcut;
- 6. If the Toolkit launches in step 5, you're all set for automatic startup.

Please see our **step-by-step guide** to learn more about how to automatically start the Toolkit after reboot.





### **6. AUTOMATIC OS UPDATES**

When running the Toolkit in a production environment, it is often desired not to have the system update and restart itself which in turn stops the Toolkit from running and analyzing camera feeds.

Settings	×
命 Home	Windows Update
Find a setting	You're up to date
Update & Security	Check for updates
${\mathbb C}$ Windows Update	Pause updates for 7 days
ഥ Delivery Optimization	Visit Advanced options to change the pause period
Windows Security	Change active hours Currently 8:00 AM to 5:00 PM
→ Backup	See update history
P Troubleshoot	Advanced options
윤 Recovery	Additional apparte controls and settings
<ul> <li>Activation</li> </ul>	See what's new
占 Find my device	Your device recently got the latest update with new features and important security improvements. Explore new features
🖁 For developers	
😸 Windows Insider Program	Looking for info on the latest updates?
🖞 Device encryption	Learn more

Previous versions of Windows allowed the user to simply disable the update service and do manual updates whenever convenient.

Since Windows 10, Microsoft has made changes in their update policy making it more difficult to stop this process.



### 6.1. Schedule downloads and OS updates

The following steps will ensure that you will be given the opportunity to schedule the download and installation of Windows updates:

- 1. Use the Windows key + R keyboard shortcut to open the **Run command**.
- 2. Type gpedit.msc and click **OK** to open the Local Group Policy Editor.

🧾 Local Group Policy Editor			- 🗆	×
<u>File</u> <u>Action</u> <u>View</u> <u>H</u> elp				
🗢 🔿 🙍 📷 🔒 📓 🖬 🛛 🔻				
<ul> <li>Store</li> <li>Store</li> <li>Sync your settings</li> <li>Tablet PC</li> <li>Task Scheduler</li> <li>Windows Calendar</li> <li>Windows Customer Experience Imp</li> <li>Windows Defender</li> <li>Windows Error Reporting</li> <li>Windows Hello for Business</li> <li>Windows Ink Workspace</li> <li>Windows Installer</li> <li>Windows Mail</li> <li>Windows Mail</li> <li>Windows Media Digital Rights Manz</li> <li>Windows Media Player</li> <li>Windows Media Player</li> <li>Windows Media Player</li> <li>Windows Resenger</li> <li>Windows Resenger</li> <li>Windows Reliability Analysis</li> <li>Windows Remote Shell</li> <li>Windows Remote Shell</li> <li>Windows Remote Shell</li> <li>Windows Remote Shell</li> <li>Windows Retirability Analysis</li> <li>Windows Retore Shell</li> <li>Windows Settings</li> <li>Windows Settings</li> <li>Windows Settings</li> <li>Windows Settings</li> </ul>	Setting Defer Windows Updates Do not display 'Install Updates and Shut Down' option in Sh Do not display 'Install Updates and Shut Down' option in Sh Do not adjust default option to 'Install Updates and Shut Do Enabling Windows Update Power Management to automati Current off auto-restart for updates during active hours Always automatically restart at the scheduled time Secify deadline before auto-restart for undate installation Configure Automatic Updates Specify infranet Microsoft update service location Automatic Updates detection frequency Remove access to use all Windows Update features Do not connect to any Windows Update Internet locations Allow non-administrators to receive update notifications Allow Automatic Updates immediate installation Turn on Software Notifications Allow Automatic Updates via Automatic Updates No auto-restart with logged on users for scheduled automat Re-prompt for restart with scheduled installations Reschedule Automatic Updates scheduled installations Allow signed updates from an intranet Microsoft update ser	State Not configured Not configured	Comment No	
<	Extended Standard			
22 setting(s)				



#### 3. Browse the following path:

Computer Configuration\Administrative Templates\Windows Components\Windows Update

- 4. On the right side, double-click the **Configure Automatic Updates** policy
- 5. On the left side, check the **Enabled** option to enable the policy

6. Under Options, you'll find a number of ways to configure automatic updates, including:

- Notify for download and notify for install.
- Auto download and notify for install.
- Auto download and schedule the install.
- Allow local admin to choose setting.

(While you can pick from any of the available options, your best choice is probably 2 - Notify for download and notify for install. This option will not only prevent the OS from downloading updates automatically, which can save you data when using a metered internet connection, but you'll get a notification when new updates are available.)

7. Click Apply

8. Click **OK** to complete the task





### 6.2. Disable OS updates

It is **not recommended to completely ignore all updates** since some security updates should be mandatory. However, to turn off the update mechanism completely you will need to alter the type of connection your system has with the internet:

- 1. Go to the Windows start menu on the bottom left and type **Settings**;
- 2. Within Settings, go to Network & Internet;
- 3. Under Status, select Change connection properties;
- 4. You will see the header **Metered connection**. If you set your connection as metered connection, Windows will treat it as a connection with **limited data usage**, preventing any updates to be downloaded;
- 5. Switch **Set as metered** connection to **On**
- 6. You're done!