

Einbinden der Wetterdaten der Davis Vantage Pro2 in das Webcam-Bild der Mobotix M22M

Als Ergänzung zu meiner Wetterstation [Vantage Pro2](#) von *Davis* nutzte ich bis 2013 zwei professionelle WEB-Kameras [M22M-IT](#) aus dem Hause [Mobotix](#). Seit Januar 2014 sind nunmehr zwei Full-HD Cams vom Typ [M25](#) im Einsatz. Die Einbindung der Wetterdaten in die beiden Kameras [M25](#) ist [hier](#) ausführlich beschrieben.

Die Wahl auf die Kameras aus dem Hause [Mobotix](#) fiel aus folgenden Gründen:

1. Outdoor Kamera (IP65)
2. PoE Versorgung der Kamera
3. Integrations-/Anzeige-Möglichkeit von Wetterdaten

Die eine Kamera ist 'gen westlichen Himmel (Blickrichtung Landsham/München) und die andere 'gen östlichen Himmel (Blickrichtung Gelting/Markt Schwaben) ausgerichtet. Die aktuellen Livebilder sind über die folgende [URL](#) erreichbar.

Konfiguration der M22M

Netzwerkmeldungen definieren

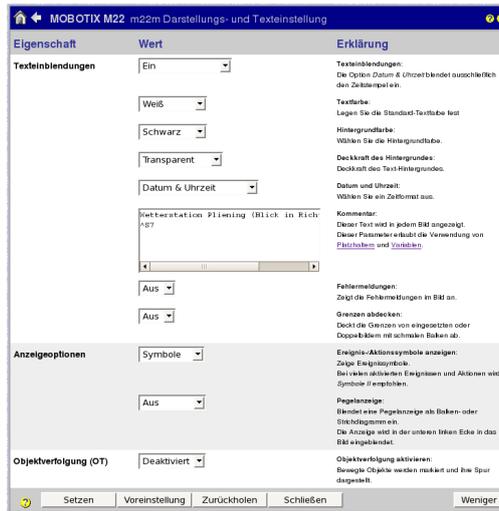
Für die Darstellung der Wetterdaten nutzen wir die Übertragungsmöglichkeiten von **Netzwerkmeldungen** bei der Kamera.

In den nachfolgend vermerkten URLs ist die **Variable {IP-der-Kamera}** entsprechend der vor befindlichen Netzwerkkonfiguration anzupassen.

Im **Setup-Menü - Bildsteuerung**, erreichbar unter

```
https://{IP-der-Kamera}/control/text
```

definieren wir nun einen Bereich, der für die spätere Texteinblendung genutzt werden soll.



Hierzu tragen wir in dem Textfeld **Kommentar** folgende Zeilen ein:

```
Name und Beschreibung der Webcam  
^S7
```

Die erste Zeile benötigt keiner großen Erklärung, sie steht für die Stationsbeschreibung.

Mit dem *Platzhalter* **^S7** definieren wir nun einen Bereich von 7 Zeilen, in denen die späteren Wetterdaten, die die Kamera via Netzwerkmeldungen erhalten soll, eingeblendet werden. Der Text wird mit weißen Buchstaben über das Kamerabild gelegt.

Netzwerkmeldungen aktivieren

Im Menüpunkt **Ereigniseinstellungen** erreichbar unter

```
https://{IP-der-Kamera}/control/event
```

aktivieren wir nun die Netzwerkmeldungen.



Den Port den wir hier festlegen, benötigen wir später für unser Shell-script welches die Übertragung zur Webcam anstoßen wird. Wichtig ist natürlich auch, dass wir die Netzwerkmeldungen an sich aktivieren, dies erfolgt mittels der Checkbox beim Menüpunkt **Netzwerkmeldung (RC)**.

Als nächstes schalten wird nun die Meldungen scharf. Über den Menüpunkt **Allgemeine**

Ereigniseinstellungen den wir über

<https://{IP-der-Kamera}/control/settings>

erreichen, aktivieren wir dann die *Scharfschaltung für Aufzeichnung, Aktionen und Meldungen.*



Zum Schluß werden die Sichern wir noch die GESAMTE Kammerakonfiguration in den permanenten Speicher der Kamera, mit der Anwahl der Schaltfläche **Schließen**. Die anschließende Sicherheitsabfrage bestätigen wir einfach mit einem „Klick auf O.K.“.

Die Konfiguration der Mobotix-Webcam ist somit abgeschlossen und wir fahren mit der Aufbereitung der erfassten Wetterdaten fort.

Datenaufbereitung mit Hilfe von wview

Mit Hilfe der html-engine unserer [wview-Installation](#) lassen wir uns einfach die gewünschten Messdaten in eine Textdatei schreiben, die wir dann später in die M22M „schieben“ und anzeigen lassen.



Diese Textdatei schreibt und dann alle 60 Sekunden der wview-Daemon in unseren Pfad auf unserem Webserver.

Das Textdatei beinhaltet somit in unserem Beispiel aktuell folgende Daten:

```
18.4°C bei 70% Feuchte
1019.7 +hPa Tendenz +
Solarstrahlung: 7 W/m^2 - UV-Index: 0.0
0 km/h aus ONO mit max. 18 km/h
Regen: 0.2 mm bei 0.0 mm / Std.
Wolkenuntergrenze: 698 m
```

Die zugehörige Vorgabedatei **/etc/wview/html/m22m-display.htx** entspricht dann folgendem Schema:

```
<!--outsideTemp-->°C bei <!--outsideHumidity-->% Feuchte
<!--barometer-->hPa Tendenz <!--baromtrend-->
Solarstrahlung: <!--solarRad--> W/m^2 - UV-Index: <!--UV-->
<!--windSpeed--> km/h aus <!--windDirectionDegrees-->° mit max. <!--
hiWindSpeed--> km/h
Regen: <!--dailyRainMM--> mm bei <!--rainRate--> mm / Std.
Wolkenuntergrenze: <!--cumulusBase--> m
```

Damit der wview-Daemon diese Datei auch alle 60 Sekunden erzeugt, tragen wir unsere **/etc/wview/html-templates.conf** die benötigte Konfiguration ein:

```
#####
###
### Robotix Texteinblendungsvorlage
#####
###
m22m-display.htx
```

Das war's schon, nun wird alle Minute die Datei mit den Texteinblendungen aktualisiert.

Damit nun diese Daten auch im Bild der Kamera auftauchen, schicken wir diese zur M22M. Hierzu schicken wir einfach den Inhalt unserer Datei **/var/www/html/vp2/m22m-display.htm** via **TCP** zum vorher definierten Port unserer Robotix-Kamera, sprich IP-Adresse/Name:

```
cat /var/www/html/vp2/m22m-display.htm >/dev/tcp/{IP-der-Kamera}/{Port}
```

Automatische Datenaktualisierung

Für die automatisierte Aktualisierung unserer Wetterdaten im WEB-Cam-Bild bedienen wir uns nun eines einfachen Shell-Scriptes, welches wir im **/bin Pfad unseres** Webcam-Users ablegen. Mit dem besten Editor unser Wahl legen wir nun also dort eine Datei **cam-update.sh** an.

```
$ vim cam-update.sh
```

```
#!/bin/bash
# Script zum Aktualisieren der Wetterdaten
# im Webcam-Bild der Mobotix M22M
# Version 0.01 vom 06.03.08

cat /var/www/html/vp2/m22m-display.htm >/dev/tcp/{IP-der-Kamera}/{Port}
```

Anschließend machen wir unser Shell-Script noch ausführbar:

```
$ chmod +x cam-update.sh
```

Und schon können wir das erste mal Testen, ob die Wetterdaten in das Webcam-Bild eingefügt werden.

```
$ ./cam-update.sh
```

Damit nun auch dieser Datenfluss jede Minute automatisch abläuft, legen wir noch einen cronjob für unseren Web-Cam-User an. Hierzu schlüpfen wir in die Rolle unseres Webcam-Users

```
su - webcam-user
```

und bearbeiten die nutzereigene **crontab**:

```
$ crontab -e
#Wetterdatenupdate der Webcam
* * * * * /home/webcam/bin/cam-update.sh
```

Das war es dann auch schon! Nun werden alle 60 Sekunden die aktuellen Wetterdaten aktualisiert und in das Bild unserer Wetter-Webcam eingefügt.

From:
<https://dokuwiki.nausch.org/> - Linux - Wissensdatenbank

Permanent link:
https://dokuwiki.nausch.org/doku.php/wetter:robotix:vantage_pro2_daten_in_robotix_m22m_einblenden

Last update: **31.10.2023 18:34.**

