COMPASS: an Interoperable Personal Health System to Monitor and Compress Signals in Chronic Obstructive Pulmonary Disease

Thomas Hofer *
Michael Schumacher *
Stefano Bromuri *
*University of Applied Sciences Western Switzerland,
Institute of Business Information Systems,
Rue de Technopole 3, Sierre, Switzerland
emails: {thomas.hofer,michaelschumacher,stefano.bromuri}@hevs.ch

***Abstract*—In the past years the progress on the mobile market has made possible an advancement in terms of telemedicine systems and definition of systems for monitoring chronic illnesses. The distribution of mobile devices in developed countries is increasing. Many of these devices are equipped with wireless standards including Bluetooth and the amount of sold Smartphones is constantly increasing. Our approach is oriented towards this market, using existing devices to enable in-home patient monitoring and even further to ubiquitious monitoring. The idea is to increase the quality of care, reduce costs and gather medical grade data, especially vital signs, with a resolution of minutes or even less, which is nowadays only possible in an ICU (Intensive Care Units). In this paper we will present the COMPASS personal health system (PHS) platform, and how it enables Android devices to collect, analyze and send sensor data to an observation storage by means of interoperability standards. We also present how this data is compressed using compressed sensing techniques and how to optimize these techniques with genetic algorithms. We also produce a preliminary evaluation of the algorithm against the state of the art algorithms for compressed sensing.**

***Keywords*—copd, compression, interoperability, soa, mobile.**

I. Introduction

Personal Health Systems [1] (PHS), systems equipped with sensors that can monitor and report on the health of a patient affected by a chronic illness, are becoming a reality, thanks to the recent advancements of mobile technology and integrated sensors. For some illnesses, such as chronic obstructive pulmonary disease (COPD), observing continuously, the evolution of the physiological value of the patient can mean being able to prevent the happening of exacerbation episodes that may force the patient to hospitalization. It is therefore of paramount importance to be able to deploy these systems on the patients to improve the interaction between the patients and the doctor.

Chronic diseases limit the quality of life drastically and are cost intensive for patients, insurance companies and governments. According to Global Health Observatory data published in [2] noncommunicable diseases (NCD), such as heart diseases, stroke, chronic respiratory diseases and diabetes, are the leading cause of mortality worldwide. Therefore the presented systems aim is to hinder this development by easily available, pervasive monitoring.

The first generation of PHS focused on simply creating interconnectivity between the sensor and a backend infrastructure to collect the data, realizing the telemedicine vision. Despite this technical advancement, a set of challenges became immediately evident: first the data collected was not structured, implying a big amount of work to the medical stuff to reconstruct the meaning of the collected data; secondly, the information to process for the medical doctors became suddenly too big and too complex to interpret; thirdly, when involving continuous sensors, the data transmitted is too much and expensive to transfer; fourthly the transmission of medical data through mobile technology brings security concerns.

Interoperability is a crucial thing when it comes to integration to either existing systems or newly created ones. Ensuring easy and fast integration can set the threshold for the success or failure of a PHS, as a matter of fact it is quite crucial to make the information immediately actionable for the medical doctors monitoring the patients. In this sense, several, major standards and protocols exist to deal with medical data on different layers. On the message layer, HL7 provides a set of standards which evolved over time. From V2.x messages in pipe and hat format to XML introducing V3 and CDA (Clinical Document Architecture) changing the paradigm from messages to documents to represent patient encounters and acts. In the presented solution the Continua Design Guidelines (CDG) were considered to ensure interoperability. This paper will discuss how the interaction between a new device like the Biovotion [1] VSM1 and the COMPASS PHS can be modelled, taking into account the CDG on different layers to maximize interoperability.

We want to introduce a flexible, extensible system which is as interoperable as possible to enable long-term monitoring with a high resolution of data for patients suffering from chronic diseases. We aim to apply information retrieval and machine learning to gain new knowledge about chronic diseases, the effects of co-morbidities and get a better understanding of the effects of certain medications.

The rest of this paper is structured as follows: Section II presents the architecture of the system; Section III discusses

[1] http://www.biovotion.ch/

PervasiveHealth 2015, May 20-23, Istanbul, Turkey

DOI 10.4108/icst.pervasivehealth.2015.259186

how the interoperability issues are tackled within the COMPASS project; in Section IV Security issues are addressed; Section V discusses the algorithms that we will use for the compression of the medical information within COMPASS and an evaluation with preliminary results; Section VI puts our work in comparison with existing PHSs; Section VII finally summarizes our approach and lists future work.

II. Architecture

The architecture of COMPASS is a server-client setting with a publish/subscribe mechanism, dynamic updates of machine learning models and RESTful services to perform CRUD (Create, Read, Update, Delete) operations. In the following sections the components shown in Figure 1 will be discussed in detail.

A. Sensor

Biovotion provides COMPASS with a multi-sensor, medical device prototype which is measuring numerous vital signs simultaneously like blood oxygenation, heart rate, temperature and triaxial accelerometer. Since no similar sensor is on the market and the earlier mentioned communication profiles aim only for a small set of devices this sensor gave the incentive to re-use existing standards and recycle them in a way to create as much interoperability as possible. The VSM1 is a wearable medical device, similar in appearance with a wristwatch-type blood pressure meter, but sleeker and more lightweight, which is placed on the bearers upper arm. The VSM1 has sensors in direct contact with the skin, which allow the continuous monitoring of vital signs like for example heartbeat, skin temperature or movement/activity. The VSM1 is intended to be a medical device class IIa (outpatient) or IIb (monitoring in a hospital) with all the respective requirements and shall be used by patients in a hospital setting, but also by mobile persons with an interest in their health. The key benefit of the VSM1 is reliable signals during rest and motion recorded in a ubiquitous manner with virtually no user intervention. Possible applications are the monitoring of patients with chronic conditions such as chronic lung diseases or obstructive sleep apnea. The hardware and communication interfaces shall allow

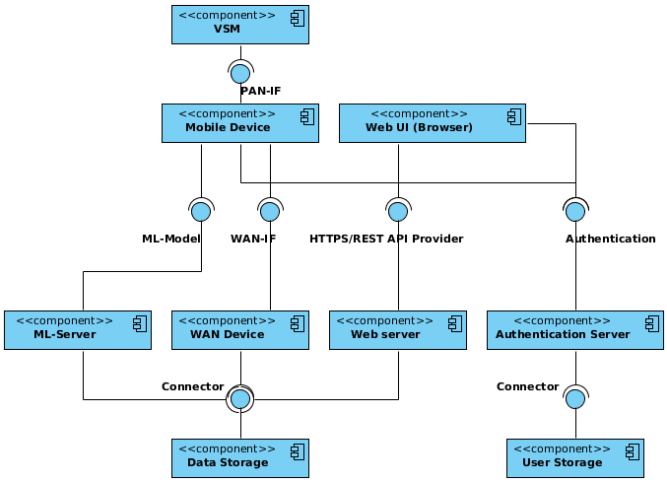

Fig. 1: Components diagram

future expansion of additional and possible external sensors (e.g. ECG) to communicate with the VSM wirelessly.

B. Mobile Device

The mobile device acts as a data collector. It has a continuous connection to the sensor, reading/receiving the measurements. At this point we should mention that the idea, architecture and system is not only designed for one particular device, it is made to be multi-device ready. The mobile device is the gateway to the server which is storing the data. It holds all relevant profiles and protocol implementations, necessary for the overall system to bring the sensors to the patients home. As a matter of fact it is planned to implement a plugin system to easily download device profile implementations. The platform of choice for the first implementation is Android 4.4 or higher since there is a security vulnerability in the keystore of the system in Android 4.3. In general Android 4.0 or higher should be used for health applications using Bluetooth because the Bluetooth Health Device Profile (HDP) was introduced in Ice Cream Sandwich (Android 4.0).

C. Authentication Server

The authentication server is part of the security as well as of the interoperability part. Offering a federated identity, using established standards, leads to interoperability and will also lead to easier integration. Offering a system which is capable of handling a federated identity scheme leads to easier integration on the server side as well as on the client side. For the authentication protocol OpenID Connect developed by the OpenID Foundation was chosen. OpenID Connect was developed with a special interest of securing RESTful services. Already implemented in server solutions, providing identity servers with different, well-known authentication schemes like SAML2, OpenID or other Single-Sign-On (SSO) solutions it improves the level of interoperability. Such servers offer different approaches with one, central user database, decoupled from the data storage which in fact increases privacy, security, and reusability of components. Another positive side effect is acceptability by other parties, dealing with established, proven security schemes.

D. API Server/WAN Device

The API server offers RESTful services for integration with the mobile device mainly. Therefore there is a strict bond between the authentication and authorization of users, logged in to the app. We claim to realize real RESTful services, meaning that every REST call can be handled as one request, no states involved. To achieve this we present in the security section a RESTful, secure approach, gathering authentication and authorization in one request.

E. Machine Learning/Predictive Component

The presented architecture implements two machine learning components. The first component is to deal with the amount of data and reduce energy consumption using compressive sensing described in Section V. The other part is a predictive component which models the submitted data to enable an alerting mechanism. The latter component is

mentioned for the sake of completeness and is part of future work.

The machine learning (ML) part resides on the backend with an information retrieval component, gathering and orchestrating the information of the retrieved measurements to apply machine learning algorithms. The ML component will calculate models and refine them on a regular basis like once per day and will be applied on both components. Figure 2 shows the schematics of the model refinement loop. Using a publish/subscribe mechanism, the server pushes the refined models to the clients if the newly created model performs better than the old one. The calculation of the models is made on the server side and the classification is done directly on the clients, which will be Android devices in the first step. Within the COMPASS project evaluations will be conducted to analyze the potential of lossy compression in the medical area.

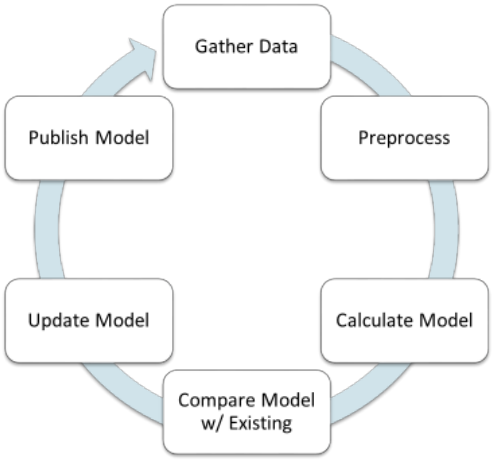

Fig. 2: Machine Learning Model Refinement Loop

F. Others

Figure 1 also shows a Web UI with a Web server and two kinds of data storage. A user storage, linked to the authentication server and a data storage linked to the WAN device or API server, the Web server and of course the Machine Learning server. These elements create a complete picture of the presented PHS but are not discussed in this paper.

III. Interoperability

Interoperability is key aspect of the presented approach and therefore we considered the Continua Design Guidelines (CDG) 2013 [3] to create an architecture with interoperable interfaces. As Bridget Moorman suitably stated in [4] about the goal of interoperability, that it *is the seamless flow of information between many disparate devices over a network.*

Due to some limitations of the hardware at hand we could not go for full Continua compliance. Continua recommends the usage of existing standards and refines them to reduce and even remove ambiguities that arise when implementing certain standards. For device communication in Body Area Networks, Continua relies on the IEEE 11073-20601 Health Informatics Application Profile - Optimized Exchange Protocol. The list of existing and well defined communication profiles in the IEEE 11073 Personal Health Device standards family and the related device specializations 104xx covers common sensors in the medical area. Building and integrating a new device with a multi-sensor approach leads to some challenges. For example one cannot rely on existing message profiles because the new device delivers a bigger or changed set of sensor values. Existing profiles have been studied and extended to fit the projects needs to keep the main patterns and rules of the standards. Furthermore, Continua references the IHE-PCD-01 (Integrated Healthcare Environment - Patient Care Devices) which mainly describes the setting between an Device Observation Reporter (DOR) and a Data Observation Customer (DOC). The message delivered by the DOR is a HL7 V2.6 message with a mapping of the IEEE 11073-10101 Nomenclature. According to the CDG the protocol ought to be a SOAP Web service to transfer the data. Interconnecting medical systems such as Hospital Information Systems (HIS), Laboratory Information Systems (LIS) or Electronic Health Record (EHR) storage, SOAP is the most common used standard but it can lead to big payloads due to the specification of certain security measures. In non-medical fields SOAP seems to be going to be replaced by RESTful services. REST [5] can be more lightweight but it is also less specified in terms of security for example. We will present a way to use existing and well known approaches to secure REST APIs and ensure message layer security, discuss the limitations and challenges that arises. Figure 3 shows a sample of a an IHE PCD HL7 message of a pulse oximenter spot measurment.

IV. Security and Privacy

We think that Security is not a feature of software, it is an essential part of it, especially when dealing with medical grade data and therefore also concerning privacy issues. We will focus on two main issues when working with mobile devices and authentication. The security of stored private/public keys on mobile devices and how to ensure the authenticity of a customer/patient in such a setting that the data is sent from the mobile device to a secured server.

A. Authentication & Authorization

The options for securing access to open interfaces are quite numerous. In the corporate and medical field most of the federated identity systems use SAML2 (Security Assertion Markup Language) assertions in combination with SOAP Web services relying on XML encryption and signing backed up by a PKI (Public Key Infrastructure.) Due to some trends and the decisions taken by big IT players influencing the development of technologies, REST [5] has become the common choice for Web services. REST is not a technology, it is a set of rules, a paradigm of how to use the HTTP protocol specification to create Web services. It is easy to create REST like services, but to consider real RESTful services, using only stateless HTTP calls, it requires a set of mechanisms, orchestrated to a well functioning system, covering authentication and authorization in a single call. OAuth has become a valid technology, but is often misused since OAuth2 is a delegation protocol which can be used for authorization and authentication purpose if it is implemented correctly. Covering authentication with

```
MSH|^~\&|Biovotion-HES-SO^1122334455667788^EUI-64||||20140505032308.221-0500||ORU^R01^ORU_R01|
    MSGID00000001|P|2.6|||NE|AL|||||IHE PCD ORU-R012006^HL7^2.16.840.1.113883.9.n.m^HL7
PID|||789567^^^Imaginary Hospital^PI||Doe^John^Joseph^^^^L
OBR|1|CESL01^Acme Mgr^1122334455667788^EUI-64|CESL01^Biovotion-HES-SO^1122334455667788^EUI
    -64|182777000^monitoring of patient^SNOMED-CT
OBX|1|CWE|68220^MDC_TIME_SYNC_PROTOCOL^MDC|0.0.0.1|532233^MDC_TIME_SYNC_GSM^MDC||||||R
OBX|2|NM|67983^MDC_ATTR_TIME_REL^MDC|0.0.0.2|5000||||||R|||20140505072308+0000||||
    ACME_Rel_Timebase-ABCDEF123456^ACME_TIMEBASE_ID
OBX|3|NM|68223^MDC_TIME_RES_REL^MDC|0.0.0.3|8000|264339^MDC_DIM_MICRO_SEC^MDC|||||R
OBX|4|CWE|68218^MDC_REG_CERT_DATA_AUTH_BODY^MDC|0.0.0.4|2^auth-body-continua||||||R
OBX|5|ST|588800^MDC_REG_CERT_DATA_CONTINUA_VERSION^MDC|0.0.0.5|1.5||||||R
OBX|6||528388^MDC_DEV_SPEC_PROFILE_PULS_OXIM^MDC|1|||||||X|||||||8877665544332211 99^EUI-64
OBX|7|ST|531970^MDC_ID_MODEL_MANUFACTURER^MDC|1.0.0.1|PulseOximeter||||||R
OBX|8|ST|531969^MDC_ID_MODEL_NUMBER^MDC|1.0.0.2|Nonin Onyx II||||||R
OBX|9|ST|531972^MDC_ID_PROD_SPEC_SERIAL^MDC|1.0.0.3|1234||||||R|||||||SN10404^EUI-64
OBX|10|NM|150456^MDC_PULS_OXIM_SAT_O2^MDC|1.0.0.4|90|262688^MDC_DIM_PERCENT^MDC|||||R
OBX|11|NM|149530^MDC_PULS_OXIM_PULS_RATE^MDC|1.0.0.5|51|264864^MDC_DIM_BEAT_PER_MIN^MDC|||||R
```

Fig. 3: HL7 V2.6 message using the IHE PCD profile of a pulse oximeter (IEEE 11073-10404 [6]) spot measurement.

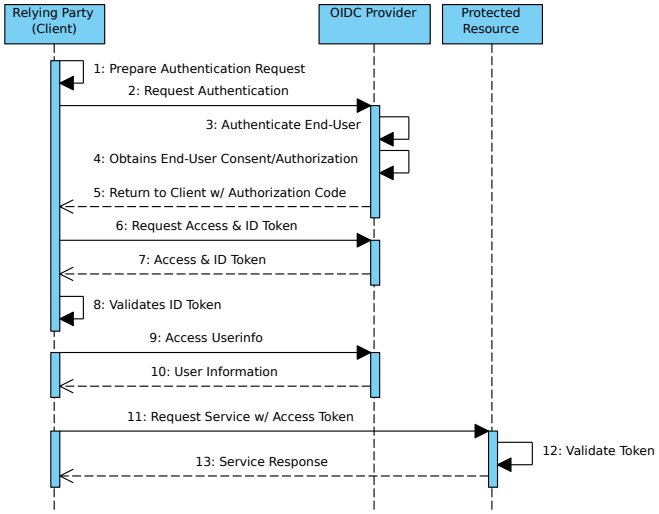

Fig. 4: OpenID Connect authentication scheme

SSO capabilities, OpenID is a valid choice. OpenID is used to create federated identities, famous from being used by Google and Facebook. Due to this fact a new system emerged called OpenID Connect (OIDC) [7], combining OAuth2 and OpenID and is going to be widely used to secure REST services. OpenID Connect extended the concept of an access token of the OAuth specification by adding an ID token to identify the requesting entity. In OpenID Connect four roles are identified: (1) Authentication Server, (2) Relying Party (Client), (3) Protected Resource (Service Provider) and (4) End-User. The client can be any native or Web application accessing a protected resource. The protected resource in the presented scenario is a RESTful service to store and retrieve observations.

For securing the service provider the *OpenID Connect Authentication using the Authorization Code Flow* is implemented which consists of following steps described in [7]. Figure 4 illustrates the scheme which applies to OpenID Connect authorization code flows. Steps 1 to 8 show the authorization process and steps 9 to 13 complete the whole picture of requesting data from a protected resource:

1) Client prepares an Authentication Request containing the desired request parameters.
2) Client sends the request to the Authorization Server.
3) Authorization Server Authenticates the End-User.
4) Authorization Server obtains End-User Consent/-Authorization.
5) Authorization Server sends the End-User back to the Client with an Authorization Code.
6) Client requests a response using the Authorization Code at the Token Endpoint.
7) Client receives a response that contains an ID Token and Access Token in the response body.
8) Client validates the ID token and retrieves the End-User's Subject Identifier.

A central element in OAuth and OIDC is the token endpoint (step 6). It is used to retrieve an access and ID token with an authorization code which was retrieved by an earlier authorization request. Using this approach to secure a RESTful service covers authentication and authorization. If a Hash-based Message Authentication Code (HMAC) [8] is used during the authentication scheme, data integrity is covered, too. The HMAC calculation algorithm that is recommended for this scenario, basically signs the HTTP request by combining the HTTP request parameters, the timestamp, the resource URI and the access code and encrypts this set of data with a shared secret that has to be exchanged in the beginning of the communication. Summarizing the addressed security measures, the RESTful service is protected by an authentication scheme which allows to add fine grained authorization. Depending on the implementation of the identity server it is possible to grant and revoke access to a secured resource within a short time. Furthermore the connection is secured using TLS (see IV-C).

B. Public Key Infrastructure

Another approach to introduce security using encryption and signatures enabling authentication is a Public Key Infrastructure (PKI). A PKI is set of routines to establish a network

of trusted entities through a trusted Certificate Authority (CA). PKI is a well-known concept and proven over years giving a strong security scheme if the key size is chosen appropriately. The key distribution is a challenge that should not be underestimated. Topics like private key safety which describes how safe the storage is kept where the key resides or the handling of certificate revocation if a key is tampered by an adversary. In COMPASS, Android is the mobile platform of choice for the first implementations. In Android 4.3 (Jelly Bean, API Version 18) a security vulnerability described in [9] was found targeting the key store which enables code execution under the key store process. Although an adversary has to overcome some obstacles there is the chance that one could achieve to exploit this vulnerability.

1) Private Key embedded in App: Establishing a PKI within a mobile app could be achieved by embedding a keystore into the app. For example in the resource folder of the app. But this folder can be made readable to everyone so the private key is not private at all. A solution could be to use a strong password on the keystore. The next question arises, how does the user of the app receive the password on a secure channel? This approach is not recommended as long as you rely on security.

2) Generated Keypair: A possible and feasable approach which needs additional work to create a PKI also used on mobile devices is to use self-signed certificates and establish your own CA (Certificate Authority). Imagine a strategy of creating a key pair and send a Certificate Signing Request (CSR) to the CA. The creation and sending of the CSR needs some additional security because it could be target of a Man-In-The-Middle attack (MITM), replacing the CA. For example by using TLS on the CA connection storing the public key in the app the probability of an attack decreases significantly since you need to replace the public key in the app directly.

C. Transport Layer Security (TLS)

Basic transport layer security will be addressed by using TLS (formerly known as SSL). An extension of TLS is mutual TLS which allows authentication on the transport layer by using a PKI. As described before, keeping private keys on Android is delicate if you do not consider the possible vulnerabilities of the Android architecture.

V. Compression with Compressive Sensing

Compressive sensing is a sampling theory that makes use of the sparsity of signals and images to reconstruct them from incomplete information. Traditionally, an effective reconstruction of a signal has to follow the Shannon sampling theorem, stating that the sampling rate must be twice as much as the highest frequency of the signal. Most data exist in sparse form or can be made sparse by using a sparsifying basis. In such a framework, a signal with an appropriate sparse representation over a basis (or sampling matrix) is projected on a much lower dimensional space by means of a projection matrix. The original signal can then be recovered from the projection provided that the projection respects the restricted isometry property (RIP). For this reason in the literature the projection matrix is usually selected as a random projection, due to the fact that random projections have a high probability of satisfying RIP.

More formally, if we consider a vector x in some Hilbert space $\mathbb{C}^N$ and a measurement matrix Φ of dimension $M \times N$ with $M << N$, then compressive sensing foresees to project x in some coefficients y by means of Φ, as $y = \Phi x$. This projection happens by means of a sparsifying matrix $s = \Psi x$, and s is $K\text{-}sparse$. In particular it has been demonstrated that if s is $K\text{-}sparse$ then we can recover x from y by applying the following optimization problem:

$$\min_{s \in \mathbb{R}^N} ||s||_{l1} \text{ subject to } y = \Phi x = (\Phi \Psi^{-1}) s$$

In the case in which the matrix Φ respects the restricted isometry property (RIP) [10], for there is a constant δ_s defined as:

Definition 1: (Restricted Isometry Constant) For each integer $k = 1, 2 \ldots, n$ define the isometry constant δ_k of a matrix Φ as the smallest number such that $(1-\delta_k)||x||_{l2}^2 \leq ||\Phi x||_{l2}^2 \leq (1+\delta_k)||x||_{l2}^2$ holds for all vectors that are K-sparse.

If the RIP property is respected, then the projection will maintain the distance between similar vectors also in the projection space, which is important to have a good reconstruction error.

Another important concept is that of *mutual coherence*.

Definition 2: (Mutual Coherence) Given a dictionary **D=** $\Phi\Psi^{-1}$, the mutual coherence of the D is defined as: $\mu\{D\} = \max_{1 \leq i,j \leq k \text{ and } i \neq j} \frac{|d_i^T d_j|}{||d_i|| \cdot ||d_j||}$

As reported by Elad et al. in [11], mutual coherence is a measure of similarity between the columns, as if two columns are very close, then they may confuse the reconstruction process of a signal.

A. Elad and Duarte Optimization Algorithms

In [11], Elad extends the concept of t-averaged mutual coherence in order to reflect an average behaviour of the projection matrix that is more likely to perform well than the simple mutual coherence previously defined. This is defined as follows:

Definition 3: (t-averaged mutual coherence) For a dictionary **D**, its t-averaged mutual coherence is defined as the average of all absolute and normalized inner products between different columns in **D** (denoted as g_{ij}) that are above t. Formally:

$$\mu_t\{D\} = \frac{\sum_{1 \leq i,j \leq k \text{ and } i \neq j} (|g_{ij}| \geq t) \cdot |g_{ij}|}{\sum_{1 \leq i,j \leq k \text{ and } i \neq j} (|g_{ij}| \geq t)}$$

Elad suggests to minimize $\mu_t(\Phi\Psi^{-1})$ with respect to Φ. Algorithm 1 achieves this by using a threshold t and a shrinking factor γ.

Similar to Elad, Duarte-Caravajalino and Sapiro [12] defined an algorithm for the optimization of the sensing matrix and the dictionary learning based on the optimization of the mutual coherence, but without the need to iterate as in the case of Elad's algorithm. The algorithm expressed in [12] tries to find the projection matrix whose Gram matrix is as close to the identity matrix as possible, because this provides a very small mutual coherence with the dictionary matrix. To achieve this and avoid iterations, the algorithm minimizes the eigenvectors

Algorithm 1 Elad's Projection Matrix Optimization Algorithm.

The objective is to minimize $\mu_t \Phi \Psi^{-1}$
t or $t\%$ the threshold
Ψ^{-1} the dictionary
p number of measurements
γ shrinking factor
Set $\Phi_0 \in \mathbb{R}^{m \times n}$ to be any random projection matrix
for i=1:Iter **do**
  Normalize $\Phi_i \Psi^{-1}$ and obtain D_q
  Set the Gram Matrix $G_i = D_i^T D_q$
  Either use the fixed t or choose t such that $t\%$ of the off-diagonal elements in G_i are above it
  Apply Shrinking to the Gram matrix G_i
  $g_{i,j} = \begin{cases} \gamma g_{i,j} \text{ if } |g_{i,j}| \geq t \\ \gamma t \cdot sign(g_{i,j}) \text{ if } t > |g_{i,j}| \geq \gamma t \\ g_{i,j} \text{ if } \gamma t > |g_{i,j}| \end{cases}$
  Reduce the rank of G_i using SVD and force it equal to m
  Calculate $S_i = \sqrt{G_q}$
  Find Φ_{i+1} minimizing $||S_i - \Phi_i \Psi^{-1}||_F^2$
end for

of the error between the Gram matrix and the identity matrix. More details about this algorithm are discussed in [12].

B. A Genetic Algorithm to Optimize the Projection Matrix

As discussed by Elad in [11], minimizing mutual coherence is one way to improve the behaviour of the projection matrix, the issue though is that this is quite independent from the task at hand, and it can hold high reconstruction error for some signals As a consequence, optimizing a metric associated to the class of signal considered may be more appropriate. In particular we think that this is crucial in the healthcare domain in which the signal must be as close as possible to the original one. In this sense, in this paper we devise a strategy based on genetic algorithms to optimize the projection matrix. In particular we apply a genetic algorithm with an elitist approach which can be summarized in 6 steps:

1) Generate Population of N Random Matrices
2) Calculate the fit of each element of the population
3) Save the best performing matrix (elitist)
4) Combine the matrices using a crossover approach by Taking a linear combination of the crossover values
5) Mutate the Matrixes (random mutation of the numbers in the matrix)
6) Calculate fit in terms of the RMSE of the reconstructed signal wrt the original signal
7) Repeat from 3.

The input of the algorithm are the population size and the probability of mutation, where we keep fixed the compression rate to a 80% of the original signal.

C. Experimentation and Preliminary Results

For the experimentation, we used a dataset comprising of 40 SPO2 signals from 4 different individuals, captured using the VSM1 monitor. In order to find the parameters of our algorithm we split our dataset in two parts, 70% of the dataset is used for training and validation and 30% of the dataset is then used for testing by comparing the RMSE of the reconstructed signal with respect to the original signal. The signals are all compressed with 20% compression rate and we fixed K-SVD [11] as the algorithm to learn the dictionary/sampling matrix.

Fig. 5 shows an example of reconstruction using the genetic algorithm discussed in the previous section, the signals we considered in this study are 10 hours of measurements on the patient, with a distance of 5 minutes before each sampling. To perform our compression we segment our signal in a size of 25 samples, compressing to 20 values and reconstructing back to 25 samples after the transmission.

This implies that the signals have all Fig. 6 shows the GRID search performed to find the optimal parameters for the genetic algorithm. In this sense, we found that a population of 4 matrices with a crossover of 3 chromosomes produces the best results in our training dataset.

Table I shows the results of the evaluation on the test set.

Despite the fact that the result is preliminary, after the training phase, the use of an optimization approach that uses the RMSE of the reconstructed signal as the parameter for fitting the projection matrix, seems to improve in a statistically significant way the result with respect to the approaches based on mutual coherence, for the selected dataset. It is important to state that to confirm this result it is necessary to attempt a similar analysis on additional datasets, which will be the subject of future work.

VI. Related Work

Monitoring chronically ill patients by means of PHS technology has been the subject of numerous studies [13], [14], with encouraging results. Thanks to the advancements of telecommunication technology, many telemedicine systems aimed at monitoring chronic diseases have been defined in recent years, as also reported in [15]. Such a study identifies motivation for self-management, long term adherence, costs, adoption, satisfaction and outcomes as important metrics to evaluate PHSs. Since this paper presents the architecture of our system and a preliminary study concerning compression, we could not evaluate the patients' opinion about the system yet. Another important thing to say is that in general the problem of compressing the data seems to be neglected in current PHSs,

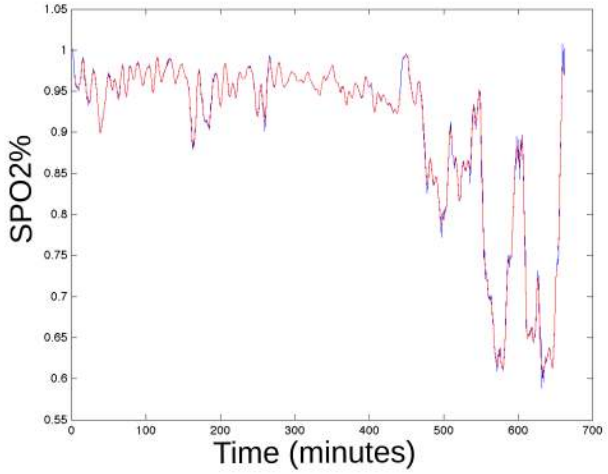

Fig. 5: Compression Results on one of the VSM1 signals. In RED the original signal, in BLUE the reconstructed one.

but this aspect can change significantly the acceptability of the patients towards the PHS, as the ability to compress the data submitted may allow to save battery time, thus reducing the discomfort of recharging the monitoring devices.

An important PHS against which comparing COMPASS is COMMODITY12 [1]. With respect to COMMODITY12, COMPASS presents an advanced handling of the models for machine learning and compression and an improved handling of interoperability by considering a CONTINUA compliant approach.

Amongst those PHS offering decision support, the most related ones are descried in the contributions presented in [16], [17], [18] and [19]. In [16], Quinn et al. present WellDoc, a mobile phone based system to provide patients with real time feedback on their glucose levels. WellDoc shows an improvement on the glucose control of the patients as compared to standard care. With respect to WellDoc, for the moment our system focuses on interoperability and compression, where the decision support part will be added in terms of prediction of exacerbation episodes in COPD.

The work of Lim et al. in [17] presents a rule based expert system producing the alerts. Similarly to [16], the system presented in [17] provides advice directly to the patient. In COMPASS, we will not use rules for the decision support, we will rather focus on producing a prediction of the physiological values of the patient.

The work of Cafazzo et al. [20] focuses on diabetes type 1. Such a contribution uses gamification patterns to monitor the patients and make them interact, showing an improvement from the perspective of the glucose control of the patients. With respect to the contribution in [20], COMPASS does not consider gamification patterns, in particular because COMPASS is focused at the level of the signals, more than at the level of the treatment. With respect to the contribution of Cafazzo et al. the main novelty of our system is to consider compressed sensing as a functionality of the PHS. The features extracted with compressed sensing will also represent the basis to produce our prediction of the SPO2 values of the patients at later stages of the COMPASS project.

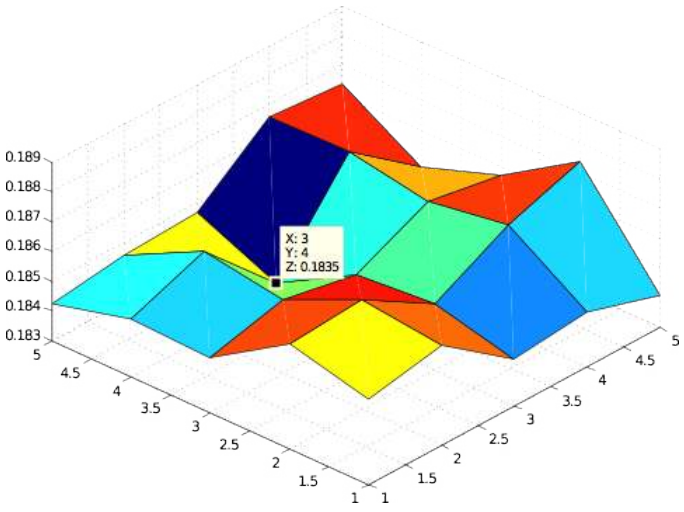

Fig. 6: Grid Search for the Parameters of the Genetic Algorithm. X: number of chromosomes for the crossover. Y: number of matrix in the population. Z: the RMSE of the reconstructed signal in the training set.

Considering related works from the broader perspective of PHSs for chronic diseases, another extensive review on the subject can be found in [21]. Given the classification of system flexibility proposed in [21], COMPASS will be modelled as a multi-function system where we offer the following services: alerting, support activities, information and documentation, analytical and diagnostic support.

The work of Tentori et al. [22] has aspects similar to our model, but the focus of their PHS is rather on applying machine learning techniques, such as Hidden Markov Models (HMM) for activity recognition. In the case of COMPASS, we have not yet built machine learning models for the task of predicting SPO2 in COPD patients, but in future work COMPASS will use features related to activity and multiple signals coming from the VSM1 sensor to predict the behaviour of SPO2 in patients affected by COPD.

In [23] the AMON project is presented. In AMON, patients affected by heart issues are monitored using a mobile solution. In AMON the monitoring is performed with an un-obtrusive device integrating several sensors in one solution. From the perspective of monitoring, AMON aimed at monitoring of multi-parametric physiological values, like we do in COMPASS, but it did not propose compression and prediction services.

VII. Conclusion & Future Work

Within this paper a solution for a Personal Health System with added value using Machine Learning techniques to introduce a lossy compression which boosts performance on the mobile device but keeps sufficient data to be of medical relevance. Furthermore, security was addressed, using state-of-the-art technologies to secure resources from the very first moment and not as an add-on. By using OpenID Connect not only security issues were covered, also the interoperability was increased in terms of reusing the mobile app with different server side implementations. Externalizing the authentication and authorization creates a certain degree of freedom to integrate the software with different solutions. For example using a sophisticated identity server providing many different authentication schemes like OAuth2, OpenID Connect or SAML2 one can use one user management and authentication system connecting with different, trusted third party application providers. Interoperability is also addressed in the means if message formats, using HL7 message format, recommended by the Continua Health Alliance.

TABLE I: RSME Comparison.

	RMSE	CI
Elad	0.1835	$\pm$ 0.0022
Duarte	0.1843	$\pm$ 0.0021
Random Matrix	0.1853	$\pm$ 0.0018
Genetic	0.1791	$\pm$ 0.0027

Future work evolving from the presented approach includes surveys for evaluating the effects of compressive sensing and the resulting discussion about acceptance in the medical field. The integration of a predictive component to implement an alerting/notification mechanism will be addressed, too. Further studies need to be conducted to evaluate the influence of an early notification of a deterioration of vital signs with regard to the monitored disease. Further work also involves the extension of the mobile application, implementing more sensor protocols extend the authorization component of the identity server with, for example, XACML [24] (eXtended Access Control Markup Language) to establish a fine-grained access management.

ACKNOWLEDGMENT

The authors would like to thank CTI (Commission for Technology and Innovation) Switzerland for funding the project COMPASS (Compass CTI 15888.1 PFES-ES) and Biovotion for providing us with their VSM prototype. This work was partially supported by the FP7 287841 COMMODITY12 project.

REFERENCES

[1] Ö. Kafali, S. Bromuri, M. Sindlar, T. van der Weide, E. Aguilar-Pelaez, U. Schaechtle, B. Alves, D. Zufferey, E. Rodríguez-Villegas, M. I. Schumacher, and K. Stathis, "Commodity$_{12}$: A smart e-health environment for diabetes management," *JAISE*, vol. 5, no. 5, pp. 479–502, 2013. [Online]. Available: http://dx.doi.org/10.3233/AIS-130220

[2] World Health Organization, "Global status report on noncommunicable diseases 2010," http://whqlibdoc.who.int/publications/2011/9789240686458_eng.pdf, April 2011, online; accessed: 2015-03-10).

[3] Continua Health Alliance, "Continua Design Guidelines," December 2013.

[4] B. Moorman, "Medical device interoperability: Standards overview," 2010.

[5] R. T. Fielding and R. N. Taylor, "Principled design of the modern web architecture," in *Proceedings of the 22Nd International Conference on Software Engineering*, ser. ICSE '00. New York, NY, USA: ACM, 2000, pp. 407–416. [Online]. Available: http://doi.acm.org/10.1145/337180.337228

[6] "Health informatics-personal health device communication part 10404: Device specialization-pulse oximeter," *IEEE STD 11073-10404-2008*, pp. c1–69, 2008.

[7] N. Sakimura, J. Bradley, M. B. Jones, B. de Medeiros, and C. Mortimore, "OpenID Connect Core 1.0 incorporating errata set 1," http://openid.net/specs/openid-connect-core-1_0.html, The OpenID Foundation, specification, 2014, online; accessed: 2015-03-10.

[8] M. Bellare, R. Canetti, and H. Krawczyk, "Keying hash functions for message authentication." in *CRYPTO*, ser. Lecture Notes in Computer Science, N. Koblitz, Ed., vol. 1109. Springer, 1996, pp. 1–15. [Online]. Available: http://dblp.uni-trier.de/db/conf/crypto/crypto96.html#BellareCK96

[9] R. Hay and A. Dayan, "Android keystore stack buffer overflow," 2014, http://www.exploit-db.com/docs/33864.pdf, 2015-03-15.

[10] A. M. Bruckstein, D. L. Donoho, and M. Elad, "From Sparse Solutions of Systems of Equations to Sparse Modeling of Signals and Images," *SIAM Review*, vol. 51, no. 1, pp. 34+, 2009. [Online]. Available: http://dx.doi.org/10.1137/060657704

[11] M. Elad and M. Aharon, "Image denoising via sparse and redundant representations over learned dictionaries," *IEEE Transactions on Image Processing*, vol. 15, no. 12, pp. 3736–3745, 2006. [Online]. Available: http://dx.doi.org/10.1109/TIP.2006.881969

[12] J. M. Duarte-Carvajalino and G. Sapiro, "Learning to sense sparse signals: Simultaneous sensing matrix and sparsifying dictionary optimization," *IEEE Transactions on Image Processing*, vol. 18, no. 7, pp. 1395–1408, 2009. [Online]. Available: http://dx.doi.org/10.1109/TIP.2009.2022459

[13] M. G. Dalfra, A. Nicolucci, A. Lapolla, A. Di Benedetto, G. Di Cianni, M. A. Dolci, I. Franzetti, A. Galluzzo, A. Napoli, G. Salietti, C. Santini, E. Torlone, C. Tortul, and E. Vitacolonna, "The effect of telemedicine on outcome and quality of life in pregnant women with diabetes," *J Telemed Telecare*, vol. 15, no. 5, pp. 238–242, 2009.

[14] D. S. Mastrogiannis, E. Igwe, and C. J. Homko, "The role of telemedicine in the management of the pregnancy complicated by diabetes," *Curr. Diab. Rep.*, vol. 13, no. 1, pp. 1–5, Feb 2013.

[15] O. F. El-Gayar, P. Timsina, N. Nawar, and W. Eid, "A systematic review of IT for diabetes self-management: Are we there yet?" *I. J. Medical Informatics*, vol. 82, no. 8, pp. 637–652, 2013.

[16] C. C. Quinn, S. S. Clough, J. M. Minor, D. Lender, M. C. Okafor, and A. Gruber-Baldini, "WellDoc mobile diabetes management randomized controlled trial: Change in clinical and behavioral outcomes and patient and physician satisfaction," *Diabetes Technology and Therapeutics*, vol. 10, no. 3, pp. 160–168, 2008.

[17] S. Lim, S. M. Kang, H. Shin, H. J. Lee, J. Won Yoon, S. H. Yu, S.-Y. Kim, S. Y. Yoo, H. S. Jung, K. S. Park, J. O. Ryu, and H. C. Jang, "Improved glycemic control without hypoglycemia in elderly diabetic patients using the ubiquitous healthcare service, a new medical information system," *Diabetes Care*, vol. 34, no. 2, pp. 308–313, 2011.

[18] R. Bellazzi, C. Larizza, S. Montani, A. Riva, M. Stefanelli, G. d'Annunzio, R. Lorini, E. J. Gomez, E. Hernando, E. Brugues, J. Cermeno, R. Corcoy, A. de Leiva, C. Cobelli, G. Nucci, S. Del Prato, A. Maran, E. Kilkki, and J. Tuominen, "A telemedicine support for diabetes management: the T-IDDM project," *Comput Methods Programs Biomed*, vol. 69, no. 2, pp. 147–161, Aug 2002.

[19] Y. Boukhors, R. Rabasa-Lhoret, H. Langelier, M. Soultan, A. Lacroix, and J. L. Chiasson, "The use of information technology for the management of intensive insulin therapy in type 1 diabetes mellitus," *Diabetes Metab.*, vol. 29, no. 6, pp. 619–627, Dec 2003.

[20] A. J. Cafazzo, M. Casselman, N. Hamming, K. D. Katzman, and R. M. Palmert, "Design of an mhealth app for the self-management of adolescent type 1 diabetes: A pilot study," *J Med Internet Res*, vol. 14, no. 3, p. e70, May 2012.

[21] C. Orwat, A. Graefe, and T. Faulwasser, "Towards pervasive computing in health care - A literature review," *BMC Medical Informatics and Decision Making*, vol. 8, no. 1, p. 118, 2008.

[22] M. Tentori, M. Rodriguez, and J. Favela, "An Agent-Based Middleware for the Design of Activity-Aware Applications," *IEEE Intelligent Systems*, vol. 26, no. 3, p. 1523, 2011.

[23] U. Anliker, J. A. Ward, P. Lukowicz, G. Tröster, F. Dolveck, M. Baer, F. Keita, E. B. Schenker, F. Catarsi, L. Coluccini, A. Belardinelli, D. Shklarski, M. Alon, E. Hirt, R. Schmid, and M. Vuskovic, "Amon: a wearable multiparameter medical monitoring and alert system," *IEEE Transactions on Information Technology in Biomedicine*, vol. 8, no. 4, pp. 415–427, 2004.

[24] T. Moses, "eXtensible Access Control Markup Language (XACML) Version 2.0," 2005, http://docs.oasis-open.org/xacml/2.0/access_control-xacml-2.0-core-spec-os.pdf, 2012-10-04. [Online]. Available: http://docs.oasis-open.org/xacml/2.0/access_control-xacml-2.0-core-spec-os.pdf